

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G06F 17/30

[12] 发明专利申请公开说明书

[21] 申请号 98808395.7

[43] 公开日 2001 年 7 月 4 日

[11] 公开号 CN 1302412A

[22] 申请日 1998.5.13 [21] 申请号 98808395.7

[30] 优先权

[32] 1997.7.22 [33] US [31] 08/898,652

[86] 国际申请 PCT/US98/09711 1998.5.13

[87] 国际公布 WO99/05618 英 1999.2.4

[85] 进入国家阶段日期 2000.2.22

[71] 申请人 微软公司

地址 美国华盛顿

[72] 发明人 利萨·布雷登-哈德 西蒙·H·科斯顿

威廉·B·多兰

露西·H·范德温德

[74] 专利代理机构 中国国际贸易促进委员会专利商标事务所

代理人 于 静

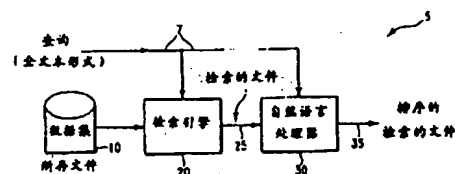
权利要求书 16 页 说明书 32 页 附图页数 14 页

[54] 发明名称 应用搜索结果的自然语言处理以改进整体精度的信息检索系统的设备和方法

[57] 摘要

应用自然语言处理以处理信息检索引擎例如基于统计的常规搜索引擎的检索结果从而改进整体精度的信息检索系统的设备和相伴方法。具体地,这类搜索最后产生一组检索文件。然后这些文件经受自然语言处理以产生一组逻辑形式。每个这类逻辑形式用“词-关系-词”方式将词组中的词之间的语义关系,具体是内容和修饰成分加以编码。以同样方式分析用户提供的查询以产生一组它们的相应的逻辑形式。按照文件和查询的逻辑形式的预定函数将文件排序。具体地,将查询的逻辑形式组和每个检索的文件的逻辑形式组比较以便确认在这两个组的任何逻辑形式之间的匹配。对每个具有至少一个匹配逻辑形式的文件探索性地计分,对匹配逻辑形式的每个不同关系赋予不同相应的预定权值。每个这类文件的分数是例如它的独一地匹配的逻辑形式的权值的预定函数。最后将留下的文件按下降分数排序并据此顺序呈现给用户。

知识产权出版社出版



ISSN 1008-4274



权 利 要 求 书

1. 一种用于从一个信息库中检索所存文件的信息检索系统中所用设备, 所述系统具有一个检索系统, 用于对一个查询作出响应而从该信息库中检索多个与该查询相关的所存文件以规定一组输出文件; 所述设备包括:

一个处理器; 及

具有存在其中的可执行指令的存储器; 及

其中该处理器对存于存储器中的指令作出响应从而:

对查询作出响应而产生一个它的第一逻辑形式, 其中第一逻辑形式描绘与该查询有关的词之间的语义关系;

为输出文件组中每个不同的文件获取一个相应的第二逻辑形式, 其中第二逻辑形式描绘所述一个文件内与一个词组有关的词之间的语义关系;

按照查询的第一逻辑形式和输出文件组中多个文件中的每一个的第二逻辑形式的预定函数来确定输出文件组中多个文件的顺序以便排序; 及

按照所排顺序提供多个与输出文件组有关的所存条目作为输出。

2. 权利要求1中的设备, 其中每个条目或是输出文件组中一个相应文件, 或是一个与所述一个相应文件有关的记录。

3. 权利要求2中的设备, 其中该查询的和输出文件组中每个不同文件的相应的第一和第二逻辑形式中的每一个是一个逻辑形式图、一个子图或一个逻辑形式三重构词表。

4. 权利要求3中的设备, 其中该处理器对所存指令作出响应而:

自一个存储媒体中读取输出文件组中的所述每一个不同文件的相应第二逻辑形式; 或

分析输出文件组中所述每一个不同文件, 从而产生所述相应的第二逻辑形式。

5. 权利要求4中的设备，其中该函数根据与该查询相关的所述第一逻辑形式及与所述一个文件相关的所述第二逻辑形式中的每一个之间的预定关系为所述一个文件计分，以及其中该处理器对所存指令作出响应，根据与输出文件中每个文件相关的分数将所存条目计分以便确定顺序。

6. 权利要求5中的设备，其中或是与该查询相关的或是与输出文件组中所述一个文件相关的所述第一或第二逻辑形式还包括分别与所述查询或所述一个文件相关的词的一个释义。

7. 权利要求6中的设备，其中所述第一和第二逻辑形式包括一个或多个逻辑形式三重构词的相应的第一和第二表，所述第一和第二表中的每一个所述逻辑形式三重构词都包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

8. 权利要求5中的设备，其中与该查询有关的所述第一逻辑形式及与输出文件组中任何文件有关的所述任何第二逻辑形式之间的匹配是一个完全的匹配。

9. 权利要求8中的设备，其中所述第一和第二逻辑形式中的每一个包括一个或多个逻辑形式三重构词的相应的第一和第二表，所述第一和第二表中的每一个所述逻辑形式三重构词都包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

10. 权利要求5中的设备，其中该信息库包括一个数据集。

11. 权利要求5中的设备，其中该查询是一个全文本查询。

12. 权利要求5中的设备，其中该检索系统是一个统计搜索引擎。

13. 权利要求5中的设备，还包括：

一个客户计算机，用于自一个用户获取一个查询及按照所述排序显示输出文件组中的多个文件；及

一个通过联网连接连至客户计算机的服务器，所述服务器包括所述处理器和所述存储器，其中该处理器对存于该存储器中的指令作出响应而：

自客户计算机中获取查询；

按所述所排顺序向客户计算机提供输出文件组中所述多个文件。

14. 权利要求13中的设备，其中服务器包括多个单独的服务器。

15. 权利要求13中的设备，其中检索系统包括一个统计搜索引擎。

16. 权利要求15中的设备，其中联网连接是一个因特网或内联网连接。

17. 权利要求16中的设备，其中该搜索引擎对查询作出响应，为输出文件组中的所述多个文件中的每一个自信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及该处理器对存于存储器中的指令和记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

18. 权利要求5中的设备还包括：

一个具有所述处理器和所述存储器的客户计算机；及

一个通过联网连接连至客户计算机的服务器，所述服务器实施所述检索系统及对由客户计算机提供的查询作出响应而向客户计算机提供所述输出文件组。

19. 权利要求18中的设备，其中该检索系统包括一个统计搜索引擎。

20. 权利要求19中的设备，其中该联网连接是一个因特网或内联网连接。

21. 权利要求20中的设备，其中该搜索引擎对查询作出响应而为输出文件组中所述多个文件中的每一个自信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及该处理器对存于存储器中的指令和记录中包含

的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

22. 权利要求5中的设备还包括一个具有所述处理器和所述存储器的计算机，其中该计算机也对存于存储器内的指令作出响应而实施所述检索系统。

23. 权利要求22中的设备，其中该检索系统包括一个统计搜索引擎。

24. 权利要求5中的设备，其中所述一个文件的分数也是所述一个文件中第二逻辑形式内的节点词，所述一个文件中所述节点词的频度或语义内容，所述一个文件中的预定义节点词的频度或语义内容，所述一个文件中特定逻辑形式三重构词的频度，或所述一个文件的长度的预定函数。

25. 权利要求24中的设备，其中该查询是一个全文本查询。

26. 权利要求24中的设备，其中该检索系统包括一个统计搜索引擎。

27. 权利要求24中的设备还包括：

一个客户计算机，用于自一个用户获取一个查询，及按照所述排序显示输出文件组中多个文件；及

一个通过联网连接连至客户计算机的服务器，所述服务器包括所述处理器和所述存储器，其中该处理器对存于该存储器内的指令作出响应而：

自该客户计算机中获取该查询，及

按照所述排序向该客户计算机提供输出文件组中的所述多个文件。

28. 权利要求27中的设备，其中该服务器包括多个单独的服务器。

29. 权利要求27中的设备，其中该检索系统包括一个统计搜索引擎。

30. 权利要求29中的设备，其中该联网连接是一个因特网或内联网连接。

31. 权利要求30中的设备，其中该搜索引擎对查询作出响应而为输出文件组中所述多个文件中的每一个自该信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及该处理器对存于存储器中的指令和记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

32. 权利要求24中的设备还包括：

一个具有所述处理器和所述存储器的客户计算机；及

一个通过联网连接连至客户计算机的服务器，所述服务器实施所述检索系统及对由客户计算机提供的查询作出响应而向客户计算机提供所述输出文件组。

33. 权利要求32中的设备，其中该检索系统包括一个统计搜索引擎。

34. 权利要求33中的设备，其中该联网连接是一个因特网或内联网连接。

35. 权利要求34中的设备，其中该搜索引擎对查询作出响应而为输出文件组中所述多个文件中的每一个自该信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及该处理器对存于存储器中的指令和记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

36. 权利要求24中的设备还包括一个具有所述处理器和所述存储器的计算机，其中该计算机也对存于存储器内的指令作出响应而实施所述检索系统。

37. 权利要求36中的设备，其中该检索系统包括一个统计搜索引擎。

38. 权利要求5的中设备，其中所述第一和第二逻辑形式中的每一个包括一个或多个逻辑形式三重构词的相应的第一和第二表，所述第一和第二表中的每一个所述逻辑形式三重构词都包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文

件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

39. 权利要求38中的设备，其中或是与该查询相关的或是与输出文件组中所述一个文件相关的所述第一或第二逻辑形式三重构词还包括分别与所述查询或所述一个文件相关的词的一个释义。

40. 权利要求38中的设备，其中所述一个文件的分数也是所述一个文件中第二逻辑形式内的节点词，所述一个文件中所述节点词的频度或语义内容，所述一个文件中的预定义节点词的频度或语义内容，所述一个文件中特定逻辑形式三重构词的频度，或所述一个文件的长度的预定函数。

41. 权利要求38中的设备，其中该函数是跨越与输出文件组中所述多个文件中的每一个有关的逻辑形式三重构词而取得的权值之和，该逻辑形式三重构词及与该查询有关的至少一个逻辑形式三重构词完全匹配，其中按照与匹配的逻辑形式三重构词相关的语义关系类型赋予每个匹配的逻辑形式三重构词一个权值。

42. 权利要求41中的设备，其中该处理器对存于该存储器内的指令作出响应而：

判定与查询有关的任何逻辑形式三重构词是否和与输出文件组中任何文件有关的任何逻辑形式三重构词匹配，以便确定一个与所述任何文件有关的匹配逻辑形式三重构词；

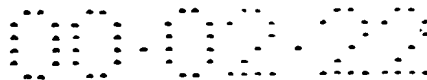
对于所述输出文件组中具有至少一个与它有关的匹配逻辑形式三重构词的每一个文件，使用由与所述每个匹配逻辑形式三重构词有关的语义关系所预定义的数字权值将所述每一个文件中的匹配逻辑形式三重构词加权，以便形成所述一个文件的一个或多个权值；

为所述一个文件计算作为所述一个或多个权值的函数的分数；
及

根据所述分数将所述文件中的每一个排序以便确定安排顺序。

43. 权利要求42中的设备，其中安排顺序是下降权值顺序。

44. 权利要求38中的设备，其中该处理器对存于存储器中的指令作出响应而为所述输出文件组中具有该文件的最高连续排序的所述输出文件组提供一个所述条目的第一预定义组。



45. 权利要求44中的设备，其中输出文件组中的多个文件包含所述输出文件组中具有至少一个与文件有关的匹配的三重构词的文件。

46. 权利要求45中的设备，其中所述第一和第二逻辑形式三重构词中的每一个包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

47. 权利要求38中的设备，其中或与查询有关或与输出文件组中所述一个文件有关的所述逻辑形式三重构词还包括一个包含所述词中的任何一个的一个超词或同音异义词的逻辑形式三重构词。

48. 权利要求38中的设备，其中与该查询有关的所述任何逻辑形式三重构词及与输出文件组中任何文件有关的所述任何逻辑形式三重构词之间的所述匹配是一个完全的匹配。

49. 权利要求38中的设备，其中该信息库包括一个数据集。

50. 权利要求38中的设备，其中该查询是一个全文本查询。

51. 权利要求38中的设备，其中该检索系统包括一个统计搜索引擎。

52. 权利要求38中的设备，还包括：

一个客户计算机，用于自一个用户获取一个查询及按照所述排序显示输出文件组中的多个文件；及

一个通过联网连接连至客户计算机的服务器，所述服务器包括所述处理器和所述存储器，其中该处理器对存于该存储器中的指令作出响应而：

自客户计算机中获取查询；及

按所述所排顺序向客户计算机提供输出文件组中所述多个文件。

53. 权利要求52中的设备，其中该服务器包括多个单独的服务器。

54. 权利要求52中的设备，其中该检索系统包括一个统计搜索引擎。

从该信息库中检索多个与该查询相关的所存文件以规定一组输出文件；该方法包括以下步骤：

对查询作出响应而产生一个它的第一逻辑形式，其中第一逻辑形式描绘与该查询有关的词之间的语义关系；

为输出文件组中每个不同的文件获取一个相应的第二逻辑形式，其中第二逻辑形式描绘所述一个文件内与一个词组有关的词之间的语义关系；

按照查询的第一逻辑形式和输出文件组中多个文件中的每一个的第二逻辑形式的预定函数来确定输出文件组中多个文件的顺序以便排序；及

按照所排顺序作为输出提供多个与输出文件组有关的所存条目。

64. 权利要求63中的方法，其中每个条目或是输出文件组中一个相应文件，或是一个与所述一个相应文件有关的记录。

65. 权利要求64中的方法，其中该查询的和输出文件组中每个不同文件的相应的第一和第二逻辑形式中的每一个是一个逻辑形式图、它的子图或一个逻辑形式三重构词表。

66. 权利要求65中的方法，其中该获取步骤包括以下步骤：

自一个存储媒体中读取输出文件组中的所述每一个不同文件的相应第二逻辑形式；或

分析输出文件组中所述每一个不同文件，从而产生所述相应的第二逻辑形式。

67. 权利要求66中的方法，其中该函数根据与该查询相关的所述第一逻辑形式及与所述一个文件相关的所述第二逻辑形式中的每一个之间的预定关系为所述一个文件计分，以及其中该排序步骤包括根据与输出文件中每个文件相关的分数将所存条目排序以便确定顺序的步骤。

68. 权利要求67中的方法，其中或是与该查询相关的或是与输出文件组中所述一个文件相关的所述第一或第二逻辑形式还包括分别与所述查询或所述一个文件相关的词的一个释义。

69. 权利要求68中的方法，其中所述第一和第二逻辑形式包括一个或多个逻辑形式三重构词的相应的第一和第二表，所述第一和第二表中的每一个所述逻辑形式三重构词都包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

70. 权利要求67中的方法，其中与该查询有关的所述第一逻辑形式及与输出文件组中任何文件有关的所述任何第二逻辑形式之间的匹配是一个完全的匹配。

71. 权利要求70中的方法，其中所述第一和第二逻辑形式中的每一个包括一个或多个逻辑形式三重构词的相应的第一和第二表，所述第一和第二表中的每一个所述逻辑形式三重构词都包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

72. 权利要求67中的方法，其中该信息库包括一个数据集。

73. 权利要求67中的方法，其中该查询是一个全文本查询。

74. 权利要求67中的方法，其中该检索系统包括一个统计搜索引擎。

75. 权利要求67中的方法，其中该系统还包括一个客户计算机，其中该方法包括在该客户计算机中的以下步骤：

自一个用户获取一个查询；及

按照所述排序显示输出文件组中的多个文件；及

该系统还包括一个通过联网连接连至客户计算机的服务器，其中该方法还包括所述服务器中的以下步骤：

自客户计算机中获取查询；及

按所述所排顺序向客户计算机提供输出文件组中所述多个文件。

76. 权利要求75中的方法，其中检索系统包括一个统计搜索引擎。

77. 权利要求76中的方法，其中联网连接是一个因特网或内联网连接。

78. 权利要求77中的方法还包括以下步骤，在该搜索引擎中对查询作出响应，为输出文件组中的所述多个文件中的每一个自信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及在该服务器中对记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

79. 权利要求67中的方法，其中该系统还包括一个客户计算机及一个通过联网连接连至客户计算机的服务器，所述服务器实施所述检索系统；其中该方法还包括在服务器中对由客户计算机提供的查询作出响应而向客户计算机提供所述输出文件组的步骤。

80. 权利要求79中的方法，其中该检索系统包括一个统计搜索引擎。

81. 权利要求80中的方法，其中该联网连接是一个因特网或内联网连接。

82. 权利要求81中的方法还包括以下步骤：在该搜索引擎中对查询作出响应而为输出文件组中所述多个文件中的每一个自信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；及在客户计算机中对记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

83. 权利要求67中的方法，其中该系统还包括一个计算机，其中该方法包括在计算机中实施所述检索系统的步骤。

84. 权利要求83中的方法，其中该检索系统包括一个统计搜索引擎。

85. 权利要求67中的方法，其中所述一个文件的分数也是所述一个文件中第二逻辑形式内的节点词，所述一个文件中所述节点词的频度或语义内容，所述一个文件中的预定义节点词的频度或语义内容，所述一个文件中特定逻辑形式三重构词的频度，或所述一个文件的长度的预定函数。

86. 权利要求85中的方法，其中该信息库包括一个数据集。

87. 权利要求85中的方法，其中该查询是一个全文本查询。

88. 权利要求85中的方法，其中该检索系统包括一个统计搜索引擎。

89. 权利要求85中的方法，其中该系统还包括一个客户计算机，其中该方法包括在客户计算机中的以下步骤：

自一个用户获取一个查询，及

按照所述排序显示输出文件组中多个文件；及

该系统还包括一个通过联网连接连至客户计算机的服务器，其中该方法还包括在所述服务器中的以下步骤：

自该客户计算机中获取该查询，及

按照所述排序向该客户计算机提供输出文件组中的所述多个文件。

90. 权利要求89中的方法，其中该检索系统包括一个统计搜索引擎。

91. 权利要求90中的方法，其中该联网连接是一个因特网或内联网连接。

92. 权利要求91中的方法，还包括以下步骤：

在该搜索引擎中对查询作出响应而为输出文件组中所述多个文件中的每一个自该信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及在服务器中对记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

93. 权利要求85中的方法，其中该系统包括一个客户计算机及一个通过联网连接连至客户计算机的服务器，所述服务器实施所述检索系统；其中该方法还包括以下步骤：在服务器中对由客户计算机提供的查询作出响应而向客户计算机提供所述输出文件组。

94. 权利要求93中的方法，其中该检索系统包括一个统计搜索引擎。

95. 权利要求94中的方法，其中该联网连接是一个因特网或内联网连接。

96. 权利要求95中的方法还包括以下步骤：在该搜索引擎中对查询作出响应而为输出文件组中所述多个文件中的每一个自该信息库中检索一个所存记录，该记录包含用于标示在哪里可以找到输出文件组中的所述每一个文件的信息；以及在该客户计算机中对记录中包含的信息作出响应，自一个与它相关的服务器中访问和下载所述每一个文件以便包括在输出文件组中。

97. 权利要求85中的方法，其中该系统还包括一个计算机，其中该方法包括在计算机中实施所述检索系统的步骤。

98. 权利要求97中的方法，其中该检索系统包括一个统计搜索引擎。

99. 权利要求67中的方法，其中所述第一和第二逻辑形式中的每一个包括一个或多个逻辑形式三重构词的相应的第一和第二表，所述第一和第二表中的每一个所述逻辑形式三重构词都包括两个词中每一个的词干形式，所述两个词分别在该查询中或在所述每一个文件中的词组中的相应逻辑形式图中在语义上相关，所述逻辑形式三重构词还包括一个表示所述两个词之间的语义关系的预定关系。

100. 权利要求99中的方法，其中或是与该查询相关的或是与输出文件组中所述一个文件相关的所述第一或第二逻辑形式三重构词还包括分别与所述查询或所述一个文件相关的词的一个释义。

101. 权利要求99中的方法，其中所述一个文件的分数也是所述一个文件中第二逻辑形式内的节点词，所述一个文件中所述节点词的频度或语义内容，所述一个文件中的预定义节点词的频度或语义内容，所述一个文件中特定逻辑形式三重构词的频度，或所述一个文件的长度的预定函数。

102. 权利要求99中的方法，其中该函数是跨越与输出文件组中所述多个文件中的每一个有关的逻辑形式三重构词而取得的权值之和，该逻辑形式三重构词及与该查询有关的至少一个逻辑形式三重构词完全匹配，其中按照与匹配的逻辑形式三重构词相关的语义关系类型赋予每个匹配的逻辑形式三重构词一个权值。

103. 权利要求102中的方法，其中排序步骤包括以下步骤：

说明书

应用搜索结果的自然语言处理以改进整体精度的信息检索系统的设备和方法

本发明涉及应用自然语言处理以处理由信息检索引擎检索的结果从而改进整体精度的信息检索系统的设备和相伴方法，该信息检索引擎是例如一个常规基于统计的搜索引擎。

自数十年前开始并延续至今的自动化信息检索技术愈来愈多地用于自海量数据库中检索存储的信息，该数据库例如包含印刷材料和/或它们的文献信息的常规数据库。这类常规数据库偏向于专门化，从而通常包含针对一个虽然广泛但却特定的题材，例如电机工程和计算机有关技术，例如由电气及电子工程师学会所维持并当今可通过例如 Knight-Ridder Information Inc. 的 Dialog Information Services 访问的 INSPEC 数据库（DIALOG 是 Knight-Ridder Information, Inc. 的注册服务商标）。当不断增多的有关文章和其他材料印刷出来时，其增长率相对地缓和和可合理地控制。此外，这类专门化的数据库组织得较好。

然而，随着可通过因特网访问的所谓“万维网”（此后简单称为“网络”）的开发和增长，以及与常规印刷相反的发送信息至网络及自其中存取信息的方便和低费用，网上可用的信息量显现出高指数的（如不是爆炸性的）增长，看上去并无实际限制。当网络在人类知识的所有领域中提供不断增长的丰富信息时，网上的信息内容是高度杂乱和极端无组织的，这使网上的信息访问和检索极其复杂和混乱。

为试图在很大程度上简化从网络检索信息的任务，在过去几年内已经开发了一系列计算机化搜索引擎以供广大公众使用。一般而言，这些常规引擎通过由软件实施的“网络爬行器”自动地访问网站和依次地跟踪其中的超文本连接并通过所谓“关键词”提取在其中遇到的每个文件并在一个大数据库中标志每个文件以备随后访问。具

体地，通过这类提取，每个由爬行器遇到的这类文件都减缩为通常所谓“词袋”，后者虽已被抽掉所有语义和句法信息，但还包含文件中具有的有内容的词。这些内容词可能存在文件本身内和/或只在该文件的超文本标记语言（HTML）版本的描述段内。在以上任何一种情况下，该引擎为每个这类文件建立一个条目即一个文件记录。对于每个文件，其内容词都在一个可搜索数据结构中加以标志，并带有一个往回指向文件记录的连接。该文件记录通常包含：（a）一个网址，即一个URL——均匀资源定位器，一个网络浏览器可通过它访问相应的文件；（b）该文件中的不同内容词以及在某些引擎中与该文件的其他内容词有关的每个这类内容词的相对地址；（c）该文件的一个短摘要，通常只是几行或该文件的前几行；及可能（d）在其HTML描述段中提供的对文件的描述。为搜索数据库，用户向引擎提供一个基于关键词的查询。该查询通常包含一个或多个用户提供的关键词，这通常只是一个由引擎容量决定的小数字，可能带有一个位于连续关键词之间的布尔型（例如“AND”或“OR”）或类似的（例如数字接近）运算符。响应于查询，该引擎试图查找包含尽可能多的关键词的文件，及如提供了一个逻辑或接近运算符，则该文件应包含所请求的关键词的特定组合或处于彼此的一定“范围”内的关键词（特定数目的关键词）。以此方式，该引擎搜索其数据库以查找包含至少一个与查询中的关键词之一匹配的词的文件，以及在有请求时根据运算符和/或由其规定的范围来查找。对于每一个它查找的这类文件，该引擎检索它的文件记录及按照该文件中相对于其他这类文件而言的关键词匹配数目来排序以向用户提供该记录。

通常，只对用户提供的关键词查询作出响应而检索的大多数文件只是与查询无关的，因而对用户无用。

因此，为减少无关的检索文件，常规基于关键词的搜索引擎（今后简单称为“统计搜索引擎”）在它们的搜索方法学中包括了统计处理。例如，根据查询中关键词与每一个检索的文件记录中的内容词之间的匹配总数以及这些关键词如何匹配，即它们是否在该组合中和/或是在一个所请求的接近范围内，统计搜索引擎为每个这类检索的文件记录计算被经常称为“统计”的数字量度。这些统计可包

括每个匹配词的反文件频度。该引擎然后按照文件记录的统计将它们排序并将一个预定小数目的，例如5-20个或更少的具有最高序数的检索文件送回至用户。一旦用户已查看了第一组检索文件的第一组文件记录（或对于某些引擎，如文件本身是由引擎送回的，则就是它们自身），则用户可请求下一组具有次高序数的文件记录，并依此类推，直至查看完所有检索的文件记录。

传统上，搜索引擎的性能是按回叫和精度来评价的。作为数据集中的所有有关文件的百分比，回叫测量是对一定查询作出响应而实际上检索的这类文件的数目。另一方面，作为所有检索文件的百分比，精度测量是真正与查询有关的文件数目。我们相信在网络搜索引擎的上下文中回叫不是一个重要的性能量度，犹如最后检索的文件数不重要一样。事实上，对于有些查询，此数目可能异常地大。因此，我们相信为产生一个有用结果，并不是所有由引擎标志的有关文件都需要检索；然而，我们认为精度非常重要，也即那些具有最高序数及首先提供给用户的文件应该是与查询最有关系的。

常规统计搜索引擎的相当差的精度来源于以下假定：词是独立的变量，也即任何文字段落中词都是彼此独立地出现的。此上下文中的独立性意味着已知一个文件中出现一个词时，出现另一个词的条件概率始终为零，也即一个文件只是简单地一个无结构的词的集合或简单的“词袋”。人们可以容易地理解，对于任何语言而言，这个假定是十分错误的。如其他语言一样，英语具有一个丰富和复杂的句法和词素-语义结构，其中词的意义经常随着它们用在其中的特定语言上下文很大地变化，及上下文在任何情况下都确定一个词的意义和哪些词会随后出现。因此，出现在一个文字段落中的词并不是彼此独立的，相反它们是紧密相关的。基于关键词的搜索引擎完全忽略了这个精细的语言结构。例如，考虑一个用自然语言表达的示例性查询：“How many hearts does an octopus have?”。一个按照内容词“hearts”和“octopus”或它们的词态词干操作的统计搜索引擎可能向用户送回引导用户到一个包含一个具有其成分为以下内容词：“artichoke hearts, squid, onions and octopus”的配方的存储文件。此引擎在得到两个内容词“octopus”和“hearts”的匹配之后可能

根据例如包括接近和逻辑运算符的统计量度来确定此文件为一个很好的匹配，而实际上该文件与该查询毫不相关。

技术上有不同方案用于将句法词组的成分提取为无标记关系中的首修饰词对。然后将这些成分标志为常规统计向量空间模型中的名词（通常不带内部结构）。

这一方案的例子在J.L.Fagan的1988年Cornell University的博士论文“文件检索的自动词组标志实验：句法与非句法方法的比较”中p.i-261上有介绍。具体地说，此方案使用自然语言处理以分析英语句子和提取句法词组组成成分，其中将这些词组成分作为名词对待并使用统计向量空间模型加以标志。在检索期间，用户输入用自然语言表达的查询，在此方案中，该查询经受自然语言处理以备分析并从中提取与标志中存储的成分类似的句法词组组成成分。此后，试图把来自查询的句法词组组成成分与存储于标志中的成分相匹配。本作者将此纯粹句法方案与一个其中使用一个随机方法以识别句法词组中成分的统计方案相对照。本作者的结论是自然语言处理实质上并不比随机方案优越，以及有时候自然语言处理的确在精度上有小改进，但并不证实自然语言处理的价值。

在1996年5月6-8日Tysons Corner, Virginia的DARPA的Proceedings of Advances in Text Processing: Tipster Program Phase 2中143-148页上由T.Strzalkowski所写“自然语言信息检索：TIPSTER-2最后报告”（今后称为“DARPA报告”）以及1995年Information Processing and Management论文集的Vol.31, No.3, 397-417页上由T.Strzalkowski所写“自然语言信息检索”都描述了另一个基于句法的方案，它使用自然语言处理以便选择合适名词以包括在搜索查询中。虽然此方案提供理论意义，但该作者在DARPA报告的147-8页上得出结论：由于实施基本自然语言技术需要复杂处理，因此该方案不实际，原文如下：

“...重要的是记住能满足我们性能要求（或至少能接近此要求）的NLP[自然语言处理]技术在处理自然语言文本中的能力仍然相当不足。特别是，涉及概念结构，逻辑形式等的先进处理仍然在计算上不能达到要求。可以假定这些先进技术

将证明为更加有效，因为它们涉及表示层限制的问题；然而，实验证据不充分并且只能限于较小范围实验”。

在1997年6月25 - 27日加拿大魁北克 McGill University 的 Conference Proceedings of RIAO97, Computer-Assisted Information Searching in Internet Vol.1, 136-155 页上 B.Katz 所写“使用自然语言为万维网注解”（今后称为“Katz 著作”）中描述了又一个这一类基于句法的方案。如 Katz 著作中所描述的，在保留内部结构的同时建立主语 - 动词 - 宾语表达式以便在检索期间容纳小的句法交替。

由于这些句法方案只得到黯淡的改进或在实施自然语言处理系统时不实用，注意力从试图直接改进查询初始结果的精度和重叫转向改进用户接口，也即具体地通过用于细化基于与用户的交互的查询的方法来改进，例如通过对于检索结果的“用户 - 类似”用户响应，还有通过显示合适群中的结果等观看查询结果的方法来改进。

虽然这些改进在它们自己方面是有用的，但通过这些改进所能得到的精度仍然过低，因此肯定不足以有效地减少关键词搜索中所固有的用户无能。具体地，仍然要求用户手动地筛选在其中只松散地分布着有关响应的相对大的文件组。

因此，需要一种技术，具体是一种设备及其相伴方法，用于检索信息从而在精度上显著地超过常规统计方案的信息检索所能得到的精度。此外，这一技术应该在任意出现的文本中的句型和长度的广阔范围内得到可靠和可重复的结果，并且在实施中可行和廉价。为显著地改进常规方案的精度和克服该技术中固有的问题，这类技术应优选地使用自然语言处理以便根据有关文件的语义内容与查询的有关内容的匹配程度来有利地选择有关文件以备检索和随后呈现给用户。

根据我们的广泛原理，本发明通过使用自然语言处理满足以下需要：改进由例如统计网络搜索引擎所完成的基于关键词的文件搜索的精度。

广义而言，此处理涉及分别与一个搜索查询和每个检索的文件相关的逻辑形式之间的匹配的生成、比较和加权。根据查询和检索文件两者的“逻辑形式”的预定函数，具体地根据与文件相关的逻辑

形式的匹配权值之和，将检索的文件排序，并最后按该排序显示文件。一个逻辑形式是一个有向无环图，其中用标记的关系来连接表示任意长度的文本的词。具体是，一个逻辑形式描绘输入串中重要词之间的语义关系，尤其是内容和修饰成分关系。此描绘可采取不同具体形式，例如一个逻辑形式图或它的任何一个子图，后者包括例如一个逻辑形式三重构词表，其中每个三重构词用“词-关系-词”形式加以阐述；其中这些形式中任何一种可用于本发明。

根据我们的特定原理，这类搜索最终从例如一个数据库或万维网中产生一组检索的文件。然后每个文件经受自然语言处理，具体是词态、句法和逻辑的形式，以备最终为每个文件的每个句子产生合适的逻辑形式。以相同的方式分析用户提供的查询以便产生一组它们的相应的逻辑形式三重构词。然后将该查询的逻辑形式组与每个所检索文件有关的逻辑形式组进行比较，以便确认查询组的逻辑形式与每个文件组的逻辑形式之间的匹配。把不产生匹配的文件消除而不再考虑。然后为每个留下的文件探索性计分。具体地，对可能在逻辑形式中出现的每个不同关系类型，也即例如深层主语、深层宾语、作用词和类似词赋予一个预定权值。每个这类留下文件的分数是其中的匹配的逻辑形式的权值的预定函数。此函数可能是例如与出现在该文件中的所有独一无二匹配三重构词（忽略双重匹配）的相关权值之和。最后，根据留下文件的分数的降序排列将它们呈现给用户，通常分为预定小数目的组，例如5或10个，用户选择时，文件自具有最高分的组开始，然后连续地随之以降序的组。

本发明可用于数个不同处理结构中：（a）基于查询和基于关键词的两种搜索（文件检索）可由一个公共计算机例如本地个人计算机（PC）处理；（b）可由一个远程计算机例如远程服务器处理基于关键词的搜索同时在例如一个客户PC上处理查询和搜索结果；或（c）可在一个客户PC上生成查询及在四处分布的不同远程服务器上进行其余处理。此外，由于数据库中每个文件在数据库中都已标志，它可预先处理以便产生可存储以供随后访问的有关逻辑形式，从而以后只要该文件被检索和经受自然语言处理时，都可节省操作时间。

结合附图阅读下列详细说明可容易地理解本发明原理，附图中：

图1阐述根据本发明的信息检索系统5的高层框图；

图2阐述图1中所示类型的使用本发明原理的信息检索系统200的高层实施例；

图3阐述包含于图2中系统200内的计算机系统300，具体是一个客户个人计算机的框图；

图4阐述图3中所示计算机300内运行的应用程序400的高层框图；

图5A - 5D阐述不同复杂程度的英语句子的不同相应例子和它们的相应逻辑形式成分；

图6阐述图6A和6B的图纸的正确对齐；

图6A和6B集合地阐述本发明检索过程600的流程图；

图7阐述过程600内运行的NLP子程序700的流程图；

图8A阐述解释性的匹配逻辑形式三重构词加权表800；

图8B图形地阐述逻辑形式三重构词比较；及阐述解释性查询和三组解释性统计检索文件的分别示于图6A和6B中的块650、660、665和670内出现的根据本发明原理的文件计分、排序和选择过程；

图9A - 9C分别阐述三个不同的实现本发明原理的信息检索系统实施例；

图9D阐述图9C中所示用于实施本发明的又一个不同实施例的远程计算机（服务器）930的迭代实施例；

图10阐述图10A和10B的图纸的正确对齐；

图10A和10B集合地阐述本发明又一个实施例，其中预先计算和存储每个文件的逻辑形式三重构词及它们的文件记录以供随后文件检索操作中访问之用；

图11阐述图10A和10B中所示文件标志引擎1015所完成的三重构词生成过程1100；

图12阐述图12A和12B的图纸的正确对齐；

图12A和12B集合地阐述图10A和10B中所示计算机系统300内所执行的本发明检索过程1200的流程图；

图13A阐述三重构词生成过程1100中运行的NLP子程序1300的流程图；及

图13B阐述检索过程1200中运行的NLP子程序1350的流程图。

为便于理解，如果可能，使用相同的参考数字以标志各图中共同的元件。

在了解下列说明后，熟悉技术的人能清楚地理解，本发明原理可容易地用于几乎任何信息检索系统以增加其中应用的搜索引擎的精度，而不论该引擎是否为一个常规引擎。此外，本发明可用于改进从几乎任何类型海量数据库中检索文字信息的精度，例如存储于磁的、光的（如CD-ROM）或其他媒体内的而不论文字信息采用何种语言，例如英语、西班牙语、德语等。

一般而言，根据本发明，我们已知道可以应用自然语言处理来处理这些记录，即最终将由其中使用的搜索引擎所提供的文件特殊地筛选和排序，从而显著地提高一个检索引擎的精度。

考虑到这点，图1阐述使用本发明的信息检索系统5的高层框图。系统5由例如基于关键词的统计检索引擎那样的常规检索引擎20及后随的处理器30所组成。处理器30使用如下所述的本发明自然语言处理技术以便将引擎20产生的文件筛选和重排序从而产生一个检索文件的有序组，后者与用户提供的查询的相关程度比其他方案都高。

具体地，运行中用户向系统5提供一个搜索查询。该查询必须具有全文本（通常称为“文字的”）形式以便通过自然语言处理充分利用其语义内容，从而提供高于单独使用引擎20时的精度。系统5将此查询应用于引擎20和处理器30两者。响应于该查询，引擎20搜索所存文件的数据集20以产生来自它们的一组检索文件。然后将此组文件（此处也称为一个“输出文件集”），如线25所标示，作为处理器30的输入量加以提供。在处理器30内，如下面详细说明的，该集中的每个文件都经受自然语言处理，尤其是词态、句法和逻辑形式，以便为该文件中每个句子产生逻辑形式。每个这类句子的逻辑形式将该句子中的语言词组中的词之间的语义关系，尤其是内容和修饰成分加以编码。处理器30以相同方式分析该查询以便产生它们的一

组相应的逻辑形式。处理器30然后将该查询的一组逻辑形式与该组中和每个文件有关的逻辑形式组比较以确认查询组中逻辑形式与每个文件的逻辑形式之间的任何匹配。没有匹配的文件即被消除而不再考虑。每个留下的包含至少一个与查询逻辑形式匹配的逻辑形式的文件即保留下来并由处理器30探索性地计分。如下面将讨论的，为每个可能在逻辑形式三重构词中出现的不同关系类型即深层主语、深层宾语、作用词和类似词赋予一个预定权值。每个这类文件的总权值（即分数）是例如所有它的独一地匹配的三重构词（即忽略双重匹配）的权值之和。最后，处理器30根据留下文件的分数向用户提供排序的文件，通常分为预定小数目的组，例如5或10个，自具有最高分的文件开始。

由于系统5非常通用和适用于广阔的不同应用范围，因此为简化以下讨论，我们将在一个解释性的上下文中讨论本发明的使用。该上下文是一个信息检索系统，它使用一个常规基于关键词的统计因特网搜索引擎以检索自万维网标志入一个数据集中的英语文件的所存记录。如下面说明的，每个这类记录通常包含一个相应文件的预定信息。对于其他搜索引擎，记录可能包含整个文件本身。虽然下面对本发明的讨论是在使用常规因特网搜索引擎的上下文中，同时该搜索引擎检索一个包含有关相应文件的一定信息的记录及该文件包括一个可找到该文件的网址，但一般而言，该引擎所检索的最终项目事实上是该文件，即使通常使用一个用到该地址的中间过程来实际地访问来自网络的文件时也是如此。在了解下面的说明后，熟悉技术的人将能容易地理解本发明如何能容易地适用于任何其他信息检索应用中。

图2阐述在一个因特网搜索引擎的上下文中使用的本发明特定实施例的高层框图。本发明主要将在此特定实施例的上下文中详细讨论。如图所示，系统200包含计算机系统300，例如客户个人计算机（PC），通过网络连接205、网络210（此处使用因特网，当然可以替代地使用任何其他这类网络例如内联网）和网络连接215连至服务器220。服务器通常包括计算机222，它装有因特网搜索引擎225并连至海量数据库227，搜索引擎225的类型例如ALTA VISTA搜索引擎

(ALTA VISTA 是 Maynard, Massachusetts 的 Digital Equipment Corporation 的注册商标), 数据库 227 通常是由搜索引擎标志的并可通过因特网上的万维网访问的文件记录的数据集。每个这类记录通常包含: (a) 一个网址 (通常称为均匀资源定位器 - URL), 网络浏览器可于该处访问相应的文件; (b) 预定义内容词, 它在某些引擎中与每一个这类词的相对于该文件中其他内容词的相对地址一起出现在该文件中; (c) 一个短摘要, 通常是该文件的几行或该文件的前几行; 及可能 (d) 如同它的超文本标记语言 (HTML) 描述段中提供的那样的文件描述。

一个在计算机系统 300 处的用户通过例如一个在此系统运行的有关的网络浏览器 (例如基于可自 Microsoft Corporation 得到的“因特网探索器”版本 3.0 浏览器并适当地修改以包括本发明原理) 建立与服务器 220 并具体地与在该处运行的搜索引擎 222 的因特网连接。此后, 用户输入一个此处标以线 201 的查询至浏览器, 后者又通过系统 300 和因特网连接将该查询送至服务器 220 和搜索引擎 225。该搜索引擎然后对于存储于数据集 227 内的文件记录处理该查询以便为由引擎确定为与查询有关的文件产生一组检索记录。鉴于引擎 225 实际上用于标志文件以形成存储于数据库 227 中的文件记录的方式以及该引擎所采取用于选择任何这类存储的文件记录的实际分析两者都与本发明无关, 我们将不再进一步讨论这两个方面。可以有把握地说, 引擎 225 对查询作出响应, 通过因特网连接将一组检索的文件记录送回至网络浏览器 420。当引擎 225 在检索文件和/或其后续者的时候, 与此同时浏览器 420 分析该查询以产生它的一组相应的逻辑形式三重构词。一旦该搜索引擎完成其搜索和已检索一组文件记录并已向浏览器提供该组文件记录, 浏览器即从相关的网络服务器中访问相应的文件本身以形成一组输出文件 (与其相关的数据库集合地形成所存储文件的一个“库房”; 这类库房也可是一个单独的数据集, 例如在一个自包含的基于 CD-ROM 的数据检索应用软件)。该浏览器 420 然后又分析每个访问的文件 (即输出文件组中的文件) 以便为每个这类文件形成一组相应的逻辑形式三重构词。此后, 如下面将详细讨论的, 根据查询和检索文件之间匹配的逻辑形式三重构词, 浏览

器420为每个具有这类匹配的文件计分并如线203所标示的根据分数的降序排列将它们呈现给用户，通常在一个预定小数目的组内，如用户通过浏览器选择，文件自具有最高分的组开始，然后连续地随之以降序的组，并依此类推，直至用户查看完足够数量的呈现的文件。虽然图2阐述了本发明利用一个网络连接以便自一个远程服务器获取文件记录 and 文件，但本发明不限于此。如下面将结合图9A详细讨论的，当检索应用软件和本发明的软件都是在一个公共计算机上例如一个本地PC上运行以及可就地访问存储于CD-ROM或其他合适的媒体内的相伴数据集时，这一网络化连接就不必要。

图3阐述示于图2中的实现本发明原理的计算机系统300的框图。

如图所示，示例为一个客户个人计算机的此系统包括输入接口（INPUT I/F）330、处理器340、通信接口（COMM I/F）350、存储器375和输出接口（OUTPUT I/F）360，全部常规地由总线370互连。通常包括不同形式设备例如示例的随机存取存储器（RAM）和硬盘存储器的存储器375（为简化起见，此处并不显示全部）中存放着操作系统（O/S）378和应用程序400。实施本发明原理的软件通常包含于应用程序400内，具体对于本实施例而言，则包含于一个网络浏览器（示于图4内）内。此操作系统可用任何常规操作系统实施，例如当今可从Redmond, Washington的Microsoft Corporation买到的WINDOWS NT操作系统（该公司也拥有注册商标“WINDOWS NT”）。由于O/S 378的组成过程与本发明无关，我们将不再讨论它。当然该浏览器及本发明的软件也可以包括于操作系统本身内。为了阐述方便和简单，我们将假定浏览器是与操作系统分开并位于应用程序400内。应用程序400在O/S 378控制下运行。对于每个包括网络浏览器在内的运行的应用程序，对每个用户规定的命令作出响应，由用户调用一个或多个单独的任务实例，这些命令通常通过用户输入设备390的可用命令选择的合适操作来交互地输入，例如通过工具条内的一个菜单或图标，然后在显示器380上显示相伴信息。

如图3所示，输入的信息可来自两条示例性外部来源：网络供应的信息，例如来自因特网和/或其他网络设施如一个内联网（全部通常表示为图2中的网络210）并通过网络连接205送至通信接口350

(示于图3)；或来自专用输入源并通过路径310送至输入接口330。专用输入可来自广泛范围的来源，例如一个或者本地或者远程的外部数据集或者其他输入源。输入接口330连至路径310及包含合适电路以便提供为物理地将每个不同专用输入信息源连接和接口至计算机系统300所需的相应电气连接。在操作系统控制下，应用程序400与外部来源，例如通过网络连接205与远程网络服务器或者通过路径310与专用源交换命令和数据，以便在程序运行期间传送和接收通常由用户请求的信息。

输入接口330也可通过连线395和用户输入设备例如键盘和鼠标电气地连至计算机系统300。显示器380例如常规彩色显示器和打印机例如常规激光打印机可分别通过连线363和367连至输出接口360。输出接口提供必要的电路以便电气地将显示器和打印机与计算机系统连接和接口。通过打印机385向用户提供来自一个运行中的应用程序的硬页输出信息。具体地，处于系统300处的用户可以通过显示器和打印机和输入设备390（主要是鼠标和键盘）的合适操作，通过因特网与包括一个可通过它访问的搜索引擎的任何一个范围广阔的远程网络服务器图像地实现通信，及从中下载信息例如文件以备就地显示和打印。

由于除用于实施本发明的所需硬件和软件外，计算机系统300的其他特定硬件部件以及存储于存储器375中的各种软件都是常规的和众所周知的，将不会再详细地讨论它们。

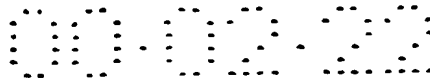
图4阐述图3中所示计算机300内运行的应用程序400的高层框图；如图4所示，在本发明范围内这些程序包括用于实施本发明的包括检索过程（这将结合图6A和6B在下面详细讨论）的网络浏览器420。假定在网络浏览器与一个用户选择的统计搜索引擎例如ALTA VISTA搜索引擎之间建立了一条因特网连接，然后用户如线422所标示地向过程600提供一个全文本（“文字的”）搜索查询。此过程如线426所示地通过网络浏览器将该查询送至搜索引擎。此外，虽未专门示出，过程600也在内部分析该查询以便产生其相应的逻辑形式三重构词，后者然后就地存储在计算机300内。对查询作出响应，该搜索引擎如线432所示地向过程600提供一组统计地检索的文件记录。如

上所述，这些记录中的每一个包括一个网址，具体是URL，可在该网址访问该文件及该文件所在远程网络服务器可请求合适命令，从而在因特网上下载一个包含该文件的计算机文件。一旦过程600接收所有记录，此过程然后如线436所示地通过网络浏览器420发送合适命令以便访问和下载所有由这些记录规定的文件（即形成输出文件组）。然后依次地从这些文件的相应网络服务器访问它们并且如线442所示地将它们下载至网络浏览器420和特定过程600。一旦下载了这些文件，过程600即分析每个这类文件以产生和就地存储它们的相应的逻辑形式三重构词。此后，将查询的逻辑形式三重构词与每个文件的逻辑形式三重构词加以比较，过程600为每个包含至少一个匹配的逻辑形式三重构词的文件计分，然后根据它们的分数将这些特定文件排序，并最后指令网络浏览器400如线446所示地在“一组再一组”基础上按照降序的文件分数向用户呈现这些特定文件。浏览器400在显示器380屏幕上生成一个合适的选择按钮（见图3），用户可恰当地在其上“点击”他（她）的鼠标以显示所需的每个连续的文件组。

为充分理解逻辑形式在确定、保存和编码语义信息中的用途，我们将在此处偏离对实施本发明的处理的讨论而在有关范围内阐释和描述本发明中使用的逻辑形式和逻辑形式三重构词并提供对用于产生它们的方式的简要了解。

广义而言，一个逻辑形式是一个有向无环图，其中用标记关系将表示任何任意长度文本的词连接起来。一个逻辑形式描绘词组中重要词之间的语义关系，它可能包括它的超名词和/或同音异义词。如图5A - 5D中将讨论和阐述的，一个逻辑形式可采取一系列不同形式中的任何一个，例如逻辑形式图或它的任何子图，例如逻辑形式三重构词表，每个三重构词具有“词 - 关系 - 词”的形式。虽然本发明如特定实施例那样生成和比较逻辑形式三重构词，但如上述，本发明可容易地利用任何其他能够描绘各词之间的语义关系的形式。

逻辑形式三重构词和它们的结果可以通过一系列逐步复杂的句子例子来更好地理解，首先考虑图5A。此图阐述示例性输入串510的



逻辑形式图515和逻辑形式三重构词525, 该句子具体是“The octopus has three hearts.”

一般而言, 为生成一个示例性输入串例如输入串510的逻辑形式三重构词, 首先将该串进行语法分析以得到它的成分词。此后, 为每个这类词使用一个存储的词典中的预定记录(不能与由搜索引擎利用的文件记录相混淆), 这些成分词的相应记录本身通过预定语法规则合并为较大结构或分析, 而它们本身又通过预定语法规则再次合并为更大结构, 例如一个句法的语法分析树。然后从该句法的语法分析树中建立一个逻辑形式图。由词记录中一定相应属性和它们的值的存在与否来部分地确定是否可对成分的特定组应用特定规则。该逻辑形式图然后转换为一系列逻辑形式三重构词。本发明的例子使用这类具有大约165,000个首词条目的词典。此词典包括不同类型的词, 例如前置词、连词、动词、名词、作用词和量词, 它们确定输入串的词中所固有的句法和语义特性从而可构作它的一个句法的语法分析树。显然, 可预先计算一个逻辑形式(或者任何其他表示, 例如任何其他能够描绘一个语义关系的逻辑形式三重构词或逻辑形式中的逻辑形式图), 而一个相应文件则在该文件的一个记录中标志和存储以备以后一旦该文件被检索时即可供随后访问和使用而不用计算。如下面结合图10-13B详细地讨论的另一个实施例中, 使用这类预先计算和存储可以显著地和有利地减少为处理任何根据本发明检索的文件所需自然语言处理量及与其相关的运行时间。

具体地, 对于一个输入串, 例如图5A中所示句子510, 首先为它的每个成分词使用词典中的预定记录进行词态分析, 以便为它们生成一个所谓“词干”(或“基干”)形式。词干形式用于将不同词形式例如动词时态和单复数名词变化规范化为一个公共的词态形式以供语法分析器用。一旦产生了词干形式, 语法分析器即使用成分词的记录中的语法规则和属性将输入串进行句法分析, 从而产生它的句法的语法分析树。此树阐述输入串的结构, 具体是输入串中每个词或词组, 例如名词词组“The octopus”的结构; 它的相应语法功能分

Dobj - - 深层宾语

Dnom - - 深层谓语主格

Dcmp - - 深层宾语补语

为标识输入串中所有语义关系，查看该串的句法的语法分析树中的每个节点。在以上关系之外，还使用其他语义作用，例如下面所示：

表3

PRED - - 谓语

PTCL - - 由两部分动词组成的助词

Ops - - 作用词，例如数词

Nadj - - 修饰名词的形容词

Dadj - - 谓语形容词

PROPS - - 是一个从句的其他没有规定的修饰语

MODS - - 不是一个从句的其他没有规定的修饰语

还定义了附加语义标记，例如：

表4

TmeAt - - 那个时候

LocAt - - 位置

在任何情况下，输入串510的这类分析结果是逻辑形式图515。输入串中彼此之间存在语义关系的那些词（例如“Octopus”和“Have”）被显示时用彼此间规定为连接属性的关系（如Dsub）彼此连接起来。由输入串510的图515所举例的这个图获取了每个输入串的内容和修饰成分的结构。其中逻辑形式分析将功能词例如前置词

和冠词映射为图中所阐述的特征或结构关系。逻辑形式分析还解决指代照应，也即规定例如一个代词和一个共同指代的名词词组之间的正确先行词关系；并检测和阐述省略的恰当功能关系。在逻辑形式分析尝试处理多义性和/或其他语言风格期间可能出现附加处理。然后简单地用常规方式自逻辑形式图中读取相应的逻辑形式三重构词并将它存储为一组。每个三重构词包含两个如图中所阐述的由一个语义关系所连接的节点词。对于示例性输入串510，逻辑形式三重构词525来自处理图515。此处逻辑形式三重构词525包含三个个别的三重构词，它们集合地表达了输入串510中固有的语义信息。

类似地，如图5B - 5D所示，对于输入串530、550和570，具体即示例性句子“The octopus has three hearts and two lungs.”，“The octopus has three hearts and it can swim.”，和“I like shark fin soup bowls.”，可分别得到逻辑形式图535、555和575以及逻辑形式三重构词540、560和580。

有三种逻辑形式构造，它们需要附加的自然语言处理以便在包括一个常规“走图”的常规方式外正确地产生所有逻辑形式三重构词，其中从逻辑形式图中建立逻辑形式三重构词。在句子并列的情况下，如在示例性句子“The octopus has three hearts and two lungs”即输入串530中，为一个词建立一个逻辑形式三重构词、它的语义关系及其并列成分的每一个值。根据“特殊”走图，我们在图540中发现两个逻辑形式三重构词“have-Dobj-heart”和“have-Dobj-lung”。只使用一个常规走图，我们只能获取一个逻辑形式三重构词“have-Dobj-and”。类似地，在具有所指事物（Refs）的成分的情况下，如在示例性句子“The octopus has three hearts and it can swim”也即输入串550中，我们在由常规走图所生成的三重构词之外还为一个词建立一个逻辑形式三重构词、它的语义关系及Refs属性的每一个值。根据此特殊走图，我们在常规逻辑形式三重构词“swim-Dsub-it”之外还在三重构词560中发现逻辑形式三重构词“swim-Dsub-octopus”。最后，在具有名词修饰语的成分的情况下，如在示例性句子“I like shark fin soup bowls”也即输入串570中，建立附加逻辑形式三重构词以表示复合名词的可能内部结构。常规走图建立逻辑形式三重构

词“bowl-Mods-shark”、“bowl-Mods-fin”和“bowl-Mods-soup”以反映可能的内部结构[[shark][fin][soup]bowl]。在特殊走图中，我们建立附加逻辑形式三重构词“fin-Mods-shark”、“soup-Mods-fin”和“soup-Mods-shark”以分别反映以下可能的内部结构 [[shark fin][soup]bowl]和 [[shark][fin soup]bowl]和 [[shark[fin]soup]bowl]。

由于词态、句法和逻辑形式的处理的特点细节与本发明无关，我们将跳过它们的任何细节。当然，关于这方面的进一步细节，读者可参考于1996年6月28日递交的名为“用于自句法树中计算语义逻辑形式的方法和系统”并赋予系列号08/674,610的共同未决美国专利申请以及可具体地参考于1997年3月7日递交的其所赋予系列号为

的“利用文本的语义表示的信息检索”；这两者都已转让给本受让人并已包括作为参考资料。

研究过逻辑形式和它们的结构后，我们将回来讨论用于实施本发明的处理操作。

用于图2、3和4中所示的本发明的特定实施例中的本发明检索过程600的流程图集合地阐述于图6A和6B中；这两张图纸的正确对齐已示于图6中。除虚线框225中所示的操作外，这些图中的其余操作都由计算机系统完成，例如客户PC300（见图2和3）及具体地在网络浏览器420内完成。为便于理解，在以下整个讨论中读者应同时参考图2、3和6A-6B。

在进入过程600时，首先进至块605。运行时此块提示用户通过网络浏览器420输入一个全文本（文字）查询。该查询可以是单个问句（例如“Are there any air-conditioned hotel sin Bali?”）或单个句子（例如“Give me contact information for all fireworks held in Seattle during the month of July.”）或句子片断（例如“Clothes in Ecuador”）。一旦获取此查询607，过程即分叉并通过路径607进至块610及通过路径643至块645。块645即调用NLP子程序700以分析该查询和结构成分并就地存储其相应的逻辑形式三重构词组。块610即如虚线615所标示地将全文本查询自网络浏览器420通过一个因特网连接传送至远程搜索引擎例如位于服务器220处的引擎225。在此处，搜索引擎完成块625以对查询作出响应而检索一组文件记录。一

式三重构词。在完全存储此组后，在块700结束执行。如要替代逻辑形式三重构词，可在本发明中使用与逻辑形式相关的一种不同表示例如一个逻辑形式图，可将块720和730容易地改变为生成作为格式化串的特定形式，而块740则用于在数据集中存储该形式以替代逻辑形式三重构词。

为充分了解本发明将逻辑形式三重构词比较和加权及将相应文件排序的方式，可考虑图8B。此图用图像阐述解释性查询和三组解释性检索文件的分别示于图6A和6B中的块650、660、665和670内出现的根据本发明原理的逻辑形式三重构词的比较；及文件计分、排序和选择过程。为便于阐述，假定用户向本发明的检索系统提供全文本查询810，其查询为“**How many hearts does an octopus have?**”。还假定，对此查询作出响应，最终通过一个统计搜索引擎将三个文件820检索。这些文件中，第一个文件（标以Document 1）是一个包含artichoke hearts和octopus的配方。第二个文件（标以Document 2）是一个有关octopi的冠词。第三个文件（标以Document 3）是一个有关deer的冠词。这三个文件和该查询都转换为它们的成分逻辑形式三重构词，该过程因此在文字上用“NLP”（自然语言处理）表示。所得该查询和Document 1、Document 2和Document 3的逻辑形式三重构词分别示于块830、840、850和860内。

一旦如此确定了这些三重构词，即如虚线845、855和865所标示，依次地将查询的逻辑形式三重构词分别与Document 1、Document 2和Document 3的逻辑形式三重构词比较，以便确认是否有任何文件包含任何与查询中任何逻辑形式三重构词相匹配的三重构词。不包含任何这类匹配的三重构词的文件例如Document 1即予以消除并不再考虑。另一方面，Document 2和Document 3包含了匹配的三重构词。具体地，Document 2包含三个这类三重构词：示例性地与一个句子相关的“HAVE-Dsub-OCTOPUS”和“HAVE-Dsub-HEART”及示例性地和另一个句子相关的“HAVE-Dsub-OCTOPUS”（这些句子并未示出）。在这些三重构词中，两个是完全相同的，即：“HAVE-Dsub-OCTOPUS”。文件的分数示例性地是该文件中所有独一地匹配的三重构词的权值的数值和。任何文件的双重匹配全

部忽略。可在一个三重构词中出现的不同类型的关系的相对加权排序按下降顺序自它们的最高分至最低分排列如下：首先是动词-宾语组合（Dobj）；然后是动词-主语组合（Dsub）；前置词和作用词（例如Ops）；及最后是修饰语（例如Nadj）。在图8A中的示例性三重构词加权表800给出这一加权方案。为简化此图，表800并未包括所有可能出现在逻辑形式三重构词中的不同关系，而只包括那些和图8B中三重构词有关系者。按此尺度，每个文件中对其分数有影响的特定三重构词都用打勾记号（“✓”）标出。当然，也可使用与我们所使用的不同的为文件计分的预定尺度，例如将权值相乘而不是相加以增强文件的选择性（区别性），或是以另外一种预定方式将权值相加，例如包括同一类型的多次匹配和/或除以上所指出的以外将其他三重构词的权值都排除。此外，对于任何文件，在某些方式中也可列为下列情况加分：该文件中三重构词本身内的节点词，或该文件中这些节点词的频度或语义内容；该文件中的特定节点词的频度或语义内容；或该文件中特定逻辑形式（或它的释义）和/或作为整体的特定逻辑形式三重构词的频度；以及该文件长度。

因此，已知我们所选用计分尺度和表800中所示权值，即可知Document2的分数是175，它由与块850中标示的文件中第一句有关的前两个三重构词的权值100和75组合而成。此块中列出的与它的第二句有关的文件中第三个三重构词早已与文件中存在的其他一个三重构词匹配，因此予以忽略。类似地，Document3的分数是100，由列于块860中的此特定文件中单个匹配三重构词的权值100组成。根据这些分数，Document2排在Document3之前，这些文件即按此顺序呈现给用户。在此处并未出现的另一种情况下，即当任何两个文件具有相同分数时，这些文件按照常规统计搜索引擎所提供的相同顺序来排序并按此序呈现给用户。

显然，熟悉技术的人知道实施本发明的不同处理部分可位于单个计算机内，也可分布在用于集合地组成一个信息检索系统的不同计算机内。在此方面，图9A-9C分别阐述三个不同的实现本发明原理的信息检索系统实施例。

图9A显示一个这类迭代实施例，其中所有处理操作都位于单个本地计算机910例如一个PC内。在此情况下，计算机910容纳一个搜索引擎并通过该引擎标志输入的文件及对用户提供的全文本查询作出响应而搜索一个数据集（或是本地的，例如在一个CD-ROM上或其他存储媒体上，或是可对该计算机访问的）以便最终产生一个组成输出文件组的检索文件组。此计算机也执行本发明以下处理：分析查询和每个这类文件两者以产生其相应的逻辑形式三重构词组；然后比较三重构词组及按照以上讨论的方式将文件选择、计分和排序，以及最后将这些结果呈现给用户，例如就在该处或由用户向该处访问。

图9B显示另一个迭代实施例，它包含图2中所示特定上下文，其中检索系统由一个与远程服务器联网的客户PC组成。此处客户PC920通过网络连接925与远程计算机（服务器）930连接。位于客户PC920处的用户输入一个全文本查询，然后PC将它在网络连接上传送至远程服务器，该客户PC还分析该查询以产生其相应的逻辑形式三重构词组。该服务器容纳例如一个常规统计搜索引擎，因此对查询作出响应而实行统计检索以产生一组文件记录。该服务器然后送回该组记录并最后或者按照客户指令或者根据搜索引擎或有关软件的能力将一组输出文件中的每个文件送回至客户PC。该客户PC然后分析它所接收的输出文件组中的相应文件以便产生它的一组逻辑形式三重构词。该客户PC然后恰当地比较两组三重构词、用上述方式将文件计分和排序，并最终将结果呈现给本地用户，从而完成其处理。

图9C中还显示又一个实施例。虽然此实施例采用了与图9B中相同的物理硬件和网络连接，但客户PC920自一个本地用户接收一个全文本查询并通过联网连接925将查询向前传送至远程计算机（服务器）930。此服务器还提供根据本发明的自然语言处理，而不是单纯地容纳一个常规搜索引擎。在此情况下，该服务器而不是该客户PC会恰当地分析该查询以产生一组相应的逻辑形式三重构词。必要时服务器也下载一组输出文件中的每个检索的文件，然后分析每个这类文件以产生它的相应的逻辑形式三重构词组。此后，服务器恰当

地比较查询和文件的两组三重构词并按照以上所述的方式将文件选择、计分和排序。排序以后，服务器930将留下的检索文件按所排顺序通过网络连接925传送至客户PC920以在该处显示。服务器可以或者按照用户的指令以上述方式在一组再一组的基础上传送这些文件，或者全部依次传送以供在它们之间按组选择并显示于客户PC上。

此外，远程计算机（服务器）930不一定只由单个用于提供如上所述的所有常规检索的和相关的自然语言处理的单个计算机实施，也可以是如图9D所示的分布处理系统而由分布于其中的单个服务器中之一承担处理操作。此处服务器930由前端处理器940组成，它通过连接950将消息发布至一系列服务器960（包含服务器1，服务器2，……，服务器n）。这些服务器中每一个实施本发明过程的一个特定部分。在这方面，服务器1可用于将输入文件标志入海量数据库上的数据集中以备随后检索。服务器2可实施一个搜索引擎，例如一个常规统计引擎，以便对一个用户提供的由前端处理器940送来的查询作出响应而自海量数据库中检索一组文件记录。这些记录将自服务器2通过前端处理器940送至例如服务器n以备随后处理，如自一个相应的网站或数据库中下载一个输出文件组中的每个相应的文件。前端处理器940也发送该查询至服务器n。服务器n然后恰当地分析该查询和每个文件以便产生相应的逻辑形式三重构词组并恰当地比较这些三重构词组以及按照上述方式将文件选择、计分和排序，然后通过前端处理器940将排序的文件送回至客户PC920以供该处排序地显示。当然，决定于运行时间和/或出现的其他条件，可以用许多任何其他方法中之一将本发明处理中不同操作分散于服务器960之间。此外，服务器930可示例性地由一个周知的系统组合配置所实施，该系统组合配置具有一个可由其中所有处理器（或其他类似的分布多处理环境）访问的分享的直接存取存储设备（DASD），它包括存储于其中的例如常规搜索引擎所用的数据库及用于自然语言处理的词典两者。

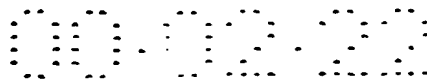
虽然我们已描述本发明是对每个检索的文件记录作出响应而下载文件和随后由例如一个客户PC就地分析该文件以产生其相应的逻辑

辑形式三重构词，但也可替代地在搜索引擎标志该文件时生成这些三重构词。在这方面，当搜索引擎通过例如使用一个网络爬行器找到并标志每个新文件时，该引擎可以为该文件下载一个完整文件，然后或立即或稍后通过一个批处理过程分析该文件及产生其逻辑形式三重构词从而预处理该文件。为完成预处理，搜索引擎在其数据库中将这三重构词存为该文件的标志记录的一部分。随后，任何时候例如对搜索查询作出响应而检索该文件记录时，即将这三重构词作为文件记录一部分送回至客户PC以供比较等用途。由于在搜索引擎中预处理文件，可以有利地各别地节省客户PC的处理时间，从而增加客户通过量。

此外，虽然我们在使用基于因特网的搜索引擎的特定环境中讨论本发明，但本发明同样可用于：（a）任何网络可访问的搜索引擎，不论它是否为基于内联网的，是否可通过专用网络设施或其他设施访问的；（b）与其所存储数据集一起运行的本地化搜索引擎，例如基于CD-ROM的数据检索应用程序，其例子是百科全书、年鉴或自包含单独数据集；和/或（c）它们的任何组合。

考虑到这些，图10A和10B集合地阐述本发明又一个实施例，其中预先处理文件以生成逻辑形式三重构词并集合地将所得三重构词、文件记录和文件本身作为自包含单独数据集存储在一个公共存储媒体上，该媒体的例子是一个或多个CD-ROM或其他可携带海量存储媒体（其例子是可移动硬盘，磁带或磁-光或大容量磁的或电子存储设备），以备分发给终端用户。图10中显示这些图纸的正确阐述。将检索应用程序本身和相伴的待搜索的数据集集合地放置于公共媒体上，即得到一个单独数据检索应用程序；因此不再需要连至远程服务器的网络连接去检索文件。

如图示，此实施例主要由三部分组成：文件标志部分1005₁，复制部分1005₂和用户部分1005₃。部分1005₁收集文件以备标志入一个数据集，例如数据集1030，它又为一个自包含文件检索应用程序形成文件信息库，例如一个百科全书，年鉴，特定图书库（例如一个决定性法律报告集）杂志装订本或类似内容。当用于复制CD-ROM和其他形式的具有相当存储容量的媒体的费用很快地下降时，



此实施例具有特殊吸引力：用于低费用地向广大用户群体传播大量收集的文件以及精确地搜索此收集文件的能力。

在任何情况下，有待标志入数据集的输入文件可自任意数目的广泛来源收集并依次地用于计算机1010。此计算机通过存于存储器1015中的恰当的软件实施一个文件标志引擎，它为每个这类文件在数据集1030内建立一个记录并将信息存入文件的记录中，以及也在数据集中建立一个包含文件本身的副本的恰当地存储的条目。引擎1015执行三重构词生成过程1100。将在下面结合图11详细讨论的这个过程是为每个被标志的文件单独执行的。此过程用主要与图6A和6B所示的块640中讨论的相同方式分析文件中的文字词组，并且如此做后在数据集1030中构作及存储该文件的一组相应的逻辑形式三重构词。由于图10A和10B所示的标志引擎1010所执行的所有其他用于标志文件的过程包括生成一个它的合适记录的过程都与本发明无关，我们将不再对它们作任何讨论。只需指出，一旦在过程1100中生成了该组三重构词，引擎1015即将此组连同文件本身的副本和建立的文件记录一起存入数据集1030。因此在完成所有标志操作后，数据集1030不单存储了其中标志的每个文件的完整副本和它的一个记录，而且存储了该文件的一组逻辑形式三重构词。

在恰当地标志了所有所需文件后，即通过复制部分1005₂把被看作“主数据集”的数据集1030本身复制。在部分1005₂内，常规媒体复制系统1040重复地将通过线1035提供的主数据集的内容的副本连同通过线1043提供的包括检索过程的检索软件 and 用户安装程序的合适软件的副本写入公共存储媒体内，例如一个或多个CD-ROM，以便集合地形成单独的文件检索应用程序。通过系统1040产生一系列媒体副本1050，它具有单独副本1050₁，1050₂，……1050_n。所有副本都是相同的，并且如副本1050₁所特定地显示的，包含一个如通过线1043所提供的文件检索应用文件那样的副本和如通过线1035所提供的数据集1030那样的副本。决定于数据集的大小和组织，每个副本可跨越一个或多个单独的媒体，例如一个或多个单独的CD-ROM。随后通常依靠一个购买的许可证将副本在用户群体内分发这些副本，如虚线1055所标示。

块1110。此块首先自该文件中提取所有文字，包括任何位于与该文件相关的HTML标记内的文字。此后，为便于执行在一个时候只对一个句子操作的自然语言处理，通过常规句子分解器将每个文件的文字分解为文字软件，其中每个句子（或问句）占据软件中的一行。此后，块1110单独为文件中的每一行文字调用NLP子程序（下面将结合图13A详细讨论它），以便分析该文件及构造该行的一组相应的逻辑形式三重构词和就地将它存储在数据集1030内。完成这些操作后，过程1100即自块1110退出。

图12A和12B集合地阐述图10A和10B中所示本发明特定实施例内所用本发明检索过程1200的流程图；其中图12阐述图12A和12B的图纸的正确对齐。与检索过程600（示于图6A和6B中及在上面讨论过）相反，图12A和12B中所示所有操作都在一个公共计算机系统上完成，此处为PC 1070（见图10A和10B）。为便于理解，读者在以下整个讨论中应同时参考图10A和10B。

进入过程1200后，首先进至块1205。此块在执行时提示用户输入一个全文本查询。在获取此查询后，过程分为两路：通过路径1207至块1210及通过路径1243至块1245。块1245在执行时调用NLP子程序1350以分析该查询并构造其相应的逻辑形式三重构词组和就地将它存储在存储器1075内。块1210在执行时如虚线1215所示，将全文本查询传送给搜索引擎1090。此处该搜索引擎完成块1220以便对查询作出响应而检索一组文件记录和与每个这类记录有关的相应的逻辑形式三重构词两者。在检索此组和相关的逻辑形式三重构词两者后，如虚线1230所示，这两者被送回至过程1200，具体是其中的块1240。块1240只是从搜索引擎1090接收此信息并将它存储在存储器1075中以便后用。虽然如上所述块1245中的操作主要是与块1210、1090和1220的操作并行的，但根据实际实施考虑，块1245的操作可以串行地在块1210、1090或1220的操作之前或后完成。

在将查询和每个检索的文件记录两者的逻辑形式三重构词组存入存储器1075后，块1250即完成。此块用以上详细讨论过的方式比较查询中每个逻辑形式三重构词及每个文件记录的每个逻辑形式三重构词，以便查找查询中任何逻辑形式三重构词与任何相应文件中

任何逻辑形式三重构词之间的匹配。完成块1250后，块1255将其中不出现匹配逻辑形式三重构词的即其逻辑形式三重构词与查询中任何逻辑形式三重构词都不匹配的文件的所有检索记录都消除。此后完成块1260。块1260如上所述地根据每个相应文件中存在的匹配逻辑形式三重构词的关系类型和它们的权值对留下的文件记录计分。在将文件记录如此加权后，完成块1265以按照分数降序将这些记录排序。最后，块1270按照所排顺序显示记录，通常分为一个预定小数目的，例如5或10个具有最高分数的文件记录组。此后用户可例如在由计算机系统1070显示的相应按钮上恰当地“点击”他（她）的鼠标以使系统显示下一组排序的文件记录，并依此类推，直至用户已充分地连续查看完所有排序的文件记录（及已访问和查看其中任何感兴趣的文件），此处过程1200结束操作并退出。

图13A阐述图11中所示三重构词生成过程1100中运行的NLP子程序1300的流程图。如上所述，NLP子程序1300分析待标志的输入文件，具体是它的单行文字，并构作该文件的一组相应的逻辑形式三重构词并在图10A和10B所示数据集1030中就地存储它。子程序1300的操作主要与图7中所示和以上详细讨论过的NLP子程序700相同。

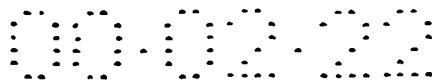
具体地，进入子程序1300后，首先执行块1310以处理一行输入文字从而产生一个逻辑形式图，例如图5A中所示示例性图515。此后，如图13A所示，块1320自图中提取（读取）一组相应的逻辑形式三重构词。在此后，块1330将每个这类逻辑形式三重构词生成成为一个单独和各别的格式化的文字串。最后块1340在数据集1030中存储该行输入文本和成为一系列格式化文字串的该行的一组逻辑形式三重构词。在完全存储该组后，过程1300退出。迭代地，如结合本发明使用另外不同形式例如一个逻辑形式图或其子图以替代逻辑形式三重构词，则可容易地修改块1320和1330以便生成该特定形式为格式化串，其中块1340在数据集中存储该形式以替代逻辑形式三重构词。

图13B阐述检索过程1200中运行的NLP子程序1350的流程图。如上所述，NLP子程序分析由User_j提供给文件检索应用程序1085（示于图10A和10B中）的查询，并构作它的一组相应的逻辑形式三重构

词及将它存储在存储器1075内。子程序1350与以上结合图13A详细讨论过的子程序1300在操作上的唯一差别在于存储相应的三重构词的位置，也即在NLP子程序1300中块1340将它存储于数据集1030中，而NLP子程序1350中块1390将它存入存储器1075中。由于子程序1350的其他块，具体是块1360、1370和1380所完成的操作主要分别与子程序1300的块1310、1320和1330所完成的操作相同，因此我们将不再讨论上述块。

为测试以上结合图1所一般描述的本发明检索过程，我们使用ALTA VISTA搜索引擎作为本检索系统的搜索引擎。此引擎可在因特网上公开访问，是一个常规统计搜索引擎，它具有超过3千1百万页标志的网页并广泛地使用着（差不多当今每天有2千8百万个调用）。我们在一台标准Pentium 90 MHz PC上实施本发明检索过程600，使用不同自然语言处理部分，包括一个词典软件，它包含于一个作为MICROSOFT OFFICE97程序包一部分的语法检查程序内（“OFFICE”和“OFFICE97”是Microsoft Corporation of Redmond, Washington的注册商标）。我们用了—个在线流水线处理模型，即当用户等待即将得到的结果时，文件被用流水线方式收集并处理。在此具体PC中，大约需要三分之一到二分之一秒以生成每个句子的逻辑形式三重构词。

请志愿者生成全文本查询以输入搜索引擎。生成了总共121条内容广泛的查询，以下是代表性的：“Why was the Celtic civilization so easily conquered by the Romans?”, “Why do antibiotics work on colds but not on viruses?”, “Who is the governor of Washington?”, “Where does the Nile cross the equator?”和“When did they start vaccinating for small pox?”. 我们将这些121条查询送至ALTA VISTA搜索引擎并在可能时获取在每个查询的响应中送回的最前面的30个文件。在这些例子中，某些文件的送回的文件少于30个，我们就使用全部送回的文件。累计下来，在121个查询中，我们获得了3361个文件（即“原始”文件）。



通过本发明的过程分析3361个文件和121个查询中的每一个以产生相应的逻辑形式三重构词组。将这些组恰当地加以比较，并用以上讨论的方式将所得文件选择、计分和排序。

手动地和个别地估价所有3361个文件与检索这些文件所用的相应查询的相关程度。为估价相关程度，我们请一位不了解我们的特定测试目的的估价人员来手动地和主观地将这些3361个文件按照它们与其相应查询的相关程度排序为“最佳”，“相关”或“不相关”。一个最佳文件可认为是包含一个对相应查询的各别答案的文件。一个相关文件是一个不包含对相应查询的各别答案但却与它相关的文件。一个不相关文件是一个对查询是无用响应的文件，例如一个与查询无关的用一种英语以外的语言的或是无法从ALTA VISTA引擎提供的相应URL中检索到的文件。为增加估价的正确性，第二位估价人员查看了这些3361个文件的子集，具体是那些具有至少一个与其相应的查询中的逻辑形式三重构词匹配的逻辑形式三重构词的文件（3361中有431个）以及那些早先安排为相关或最佳但却没有任何匹配的逻辑形式三重构词的文件（3361中有102个）。第三位估价人员作为“平局决胜员”，查看文件排序中出现的任何不一致。

我们观察到此测试的结果是，本发明检索系统实现了改进：在总精度上超过ALTA VISTA搜索引擎所送回的原始文件的精度大约200%（即所有选择的文件），即自大约16%升至大约47%，以及对于前面五个文件大约超过100%，即自大约26%升至大约51%。此外，使用本发明系统使第一个作为最佳文件送回的文件的精度比原始文件的精度提高了大约113%，自大约17%升至大约35%。

虽然我们是在统计搜索引擎的上下文中具体地讨论了本发明，但本发明不限于此。在这方面，本发明可用于处理实际上任何类型搜索引擎所获取的检索文件而改进该引擎的精度。

逻辑形式三重构词中每个不同属性的权值可以是动态的或事实上适应的而不是固定的。为做到这点，可将一个学习机理例如Bayesian或神经网络恰当地包含入本发明过程中，以便根据学习经验为每一个不同逻辑形式三重构词改变数字权值至最佳值。

虽然本发明需要逻辑形式三重构词来精确地匹配，但为在逻辑形式三重构词之间识别足够类似的语义内容，可以放宽用于匹配的准则以包含释义匹配。释义可以是词素的或是结构的。词素释义的例子是一个超名词或一个同音异义词。结构释义的例子是使用一个名词同位语或一个关系从句。例如，名词同位语结构例如“the president, Bill Clinton”应看作匹配的关系从句结构例如“Bill Clinton, who is president”。在语义水平上可以实现细度判断以确定两个词彼此之间在语义上是如何类似，从而禁止在查询“Where is coffee grown?”和文章句子例如“Coffee is frequently farmed in tropical mountainous regions.”之间出现匹配。此外，可以根据所提查询类型来修改用于判定是否存在匹配的过程。例如，如一个查询询问某个东西在哪里，则该过程应坚持将“位置”属性放在任何与所测试句子有关的逻辑形式三重构词内以便在与查询的匹配中能看到它。因此，逻辑形式三重构词“匹配”通常应规定为不单包含完全相同的匹配，也应包含从宽松的、判断性的和修改的匹配条件中引伸出来的结果。

此外，本发明可容易地与其他专门用于非文字信息例如图像、表格、视频或其他内容的处理技术结合起来，以便改进总精度。一般而言，文件中的非文字内容经常伴以语言的（文字的）描述，例如图例说明或简短说明。因此本发明过程，具体是它的自然语言处理部分，可用于分析和处理经常与非文字内容一起使用的语言描述。可使用本发明自然语言处理技术检索文件，首先查找具有在语义上与查询有关的语言内容的一组文件，然后相对于它们的非文字内容，处理此组文件以查找具有有关的文字的和非文字内容的文件。迭代地，检索文件时可以首先相对于非文字内容而检索一组文件；然后相对于它们的语言内容使用本发明技术处理该组文件以查找相关文件。

虽然此处详细地显示和说明了包含本发明原理的不同实施例，但熟悉技术的人能够容易地改变仍然使用这些原理的许多其他实施例。

说明书附图

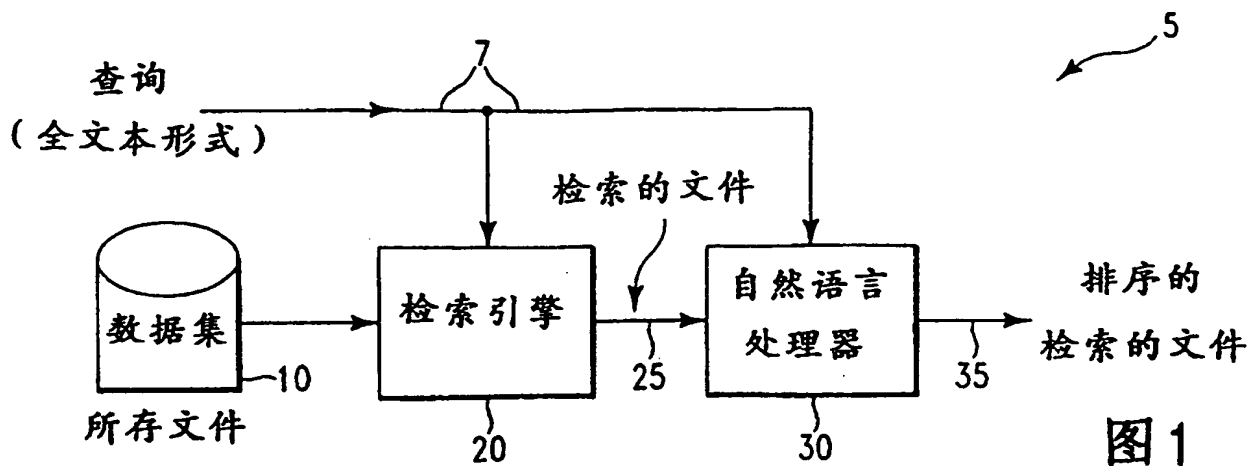


图 1

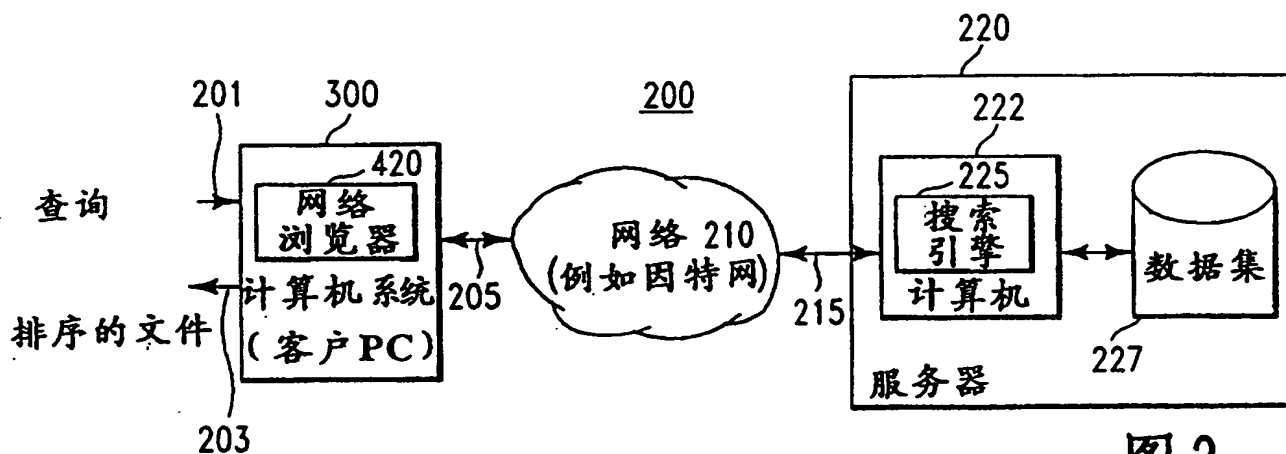


图 2

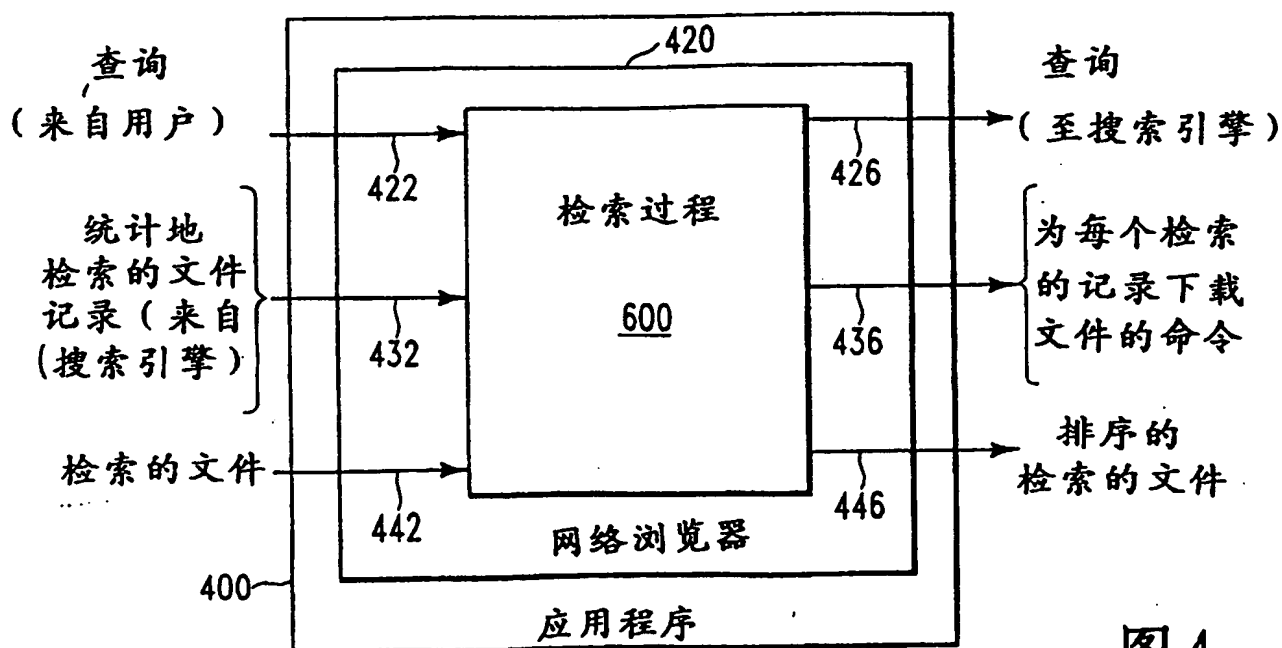


图 4

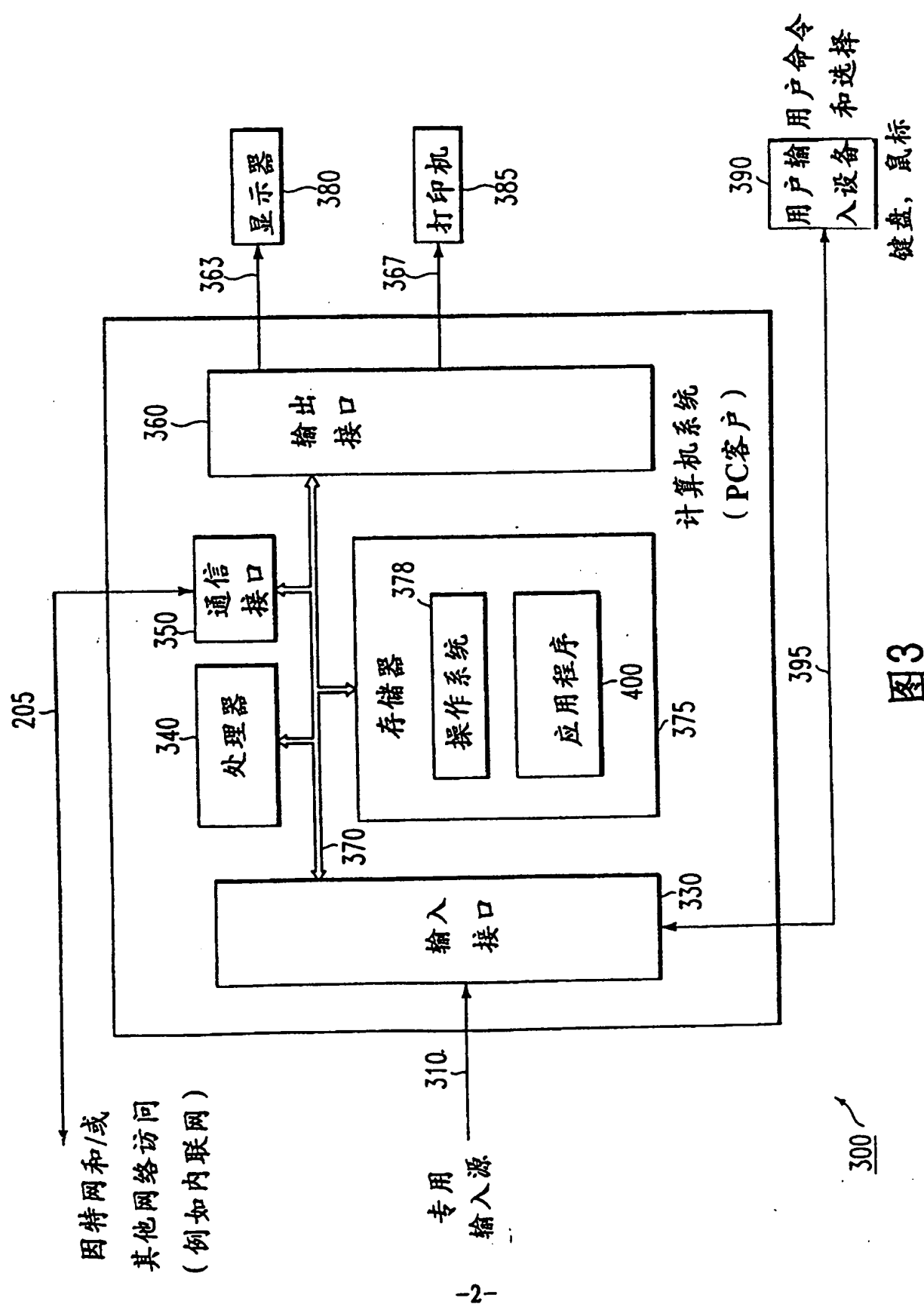
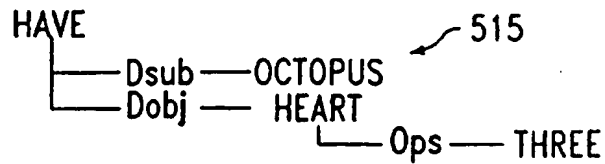


图3

510 输入串: THE OCTOPUS HAS THREE HEARTS.

逻辑形式图:



逻辑形式

三重构词:

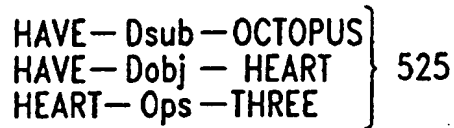
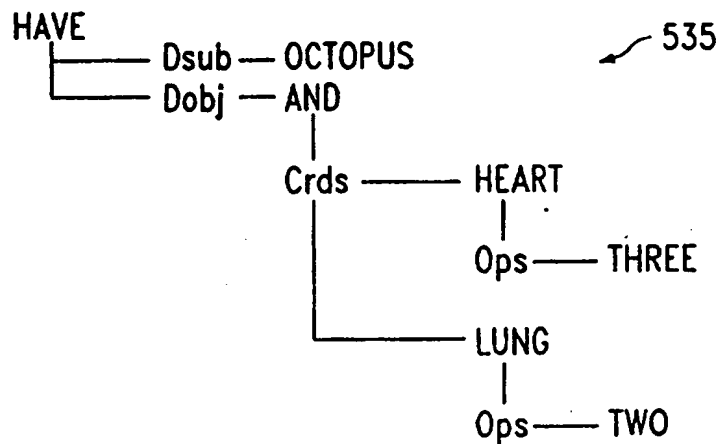


图 5A

530 输入串: THE OCTOPUS HAS THREE HEARTS AND TWO LUNGS.

逻辑形式图:



逻辑形式

三重构词:

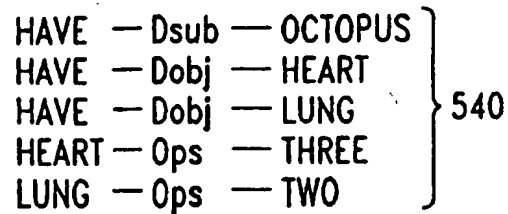


图 5B

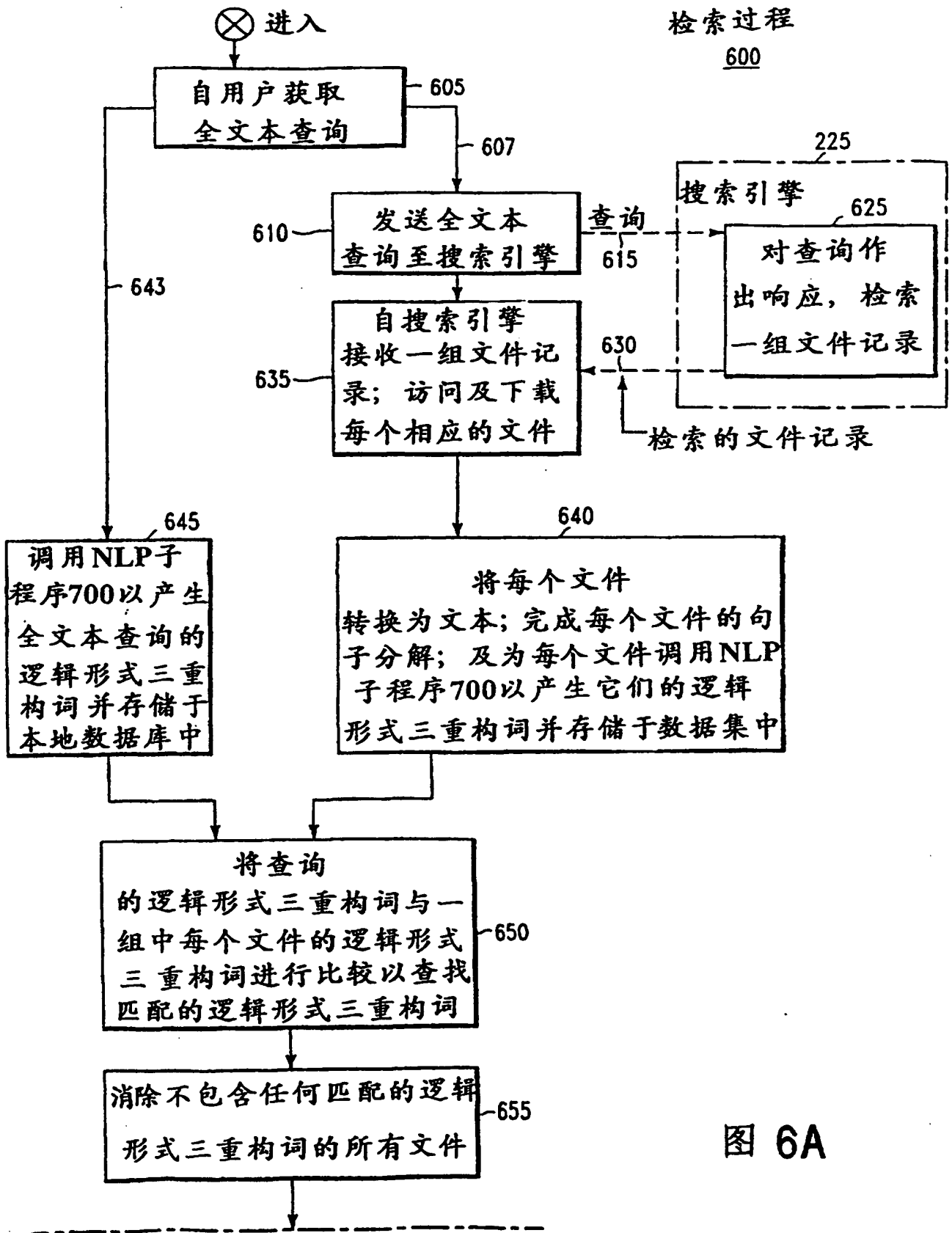


图 6A

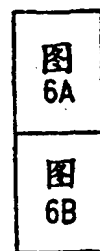
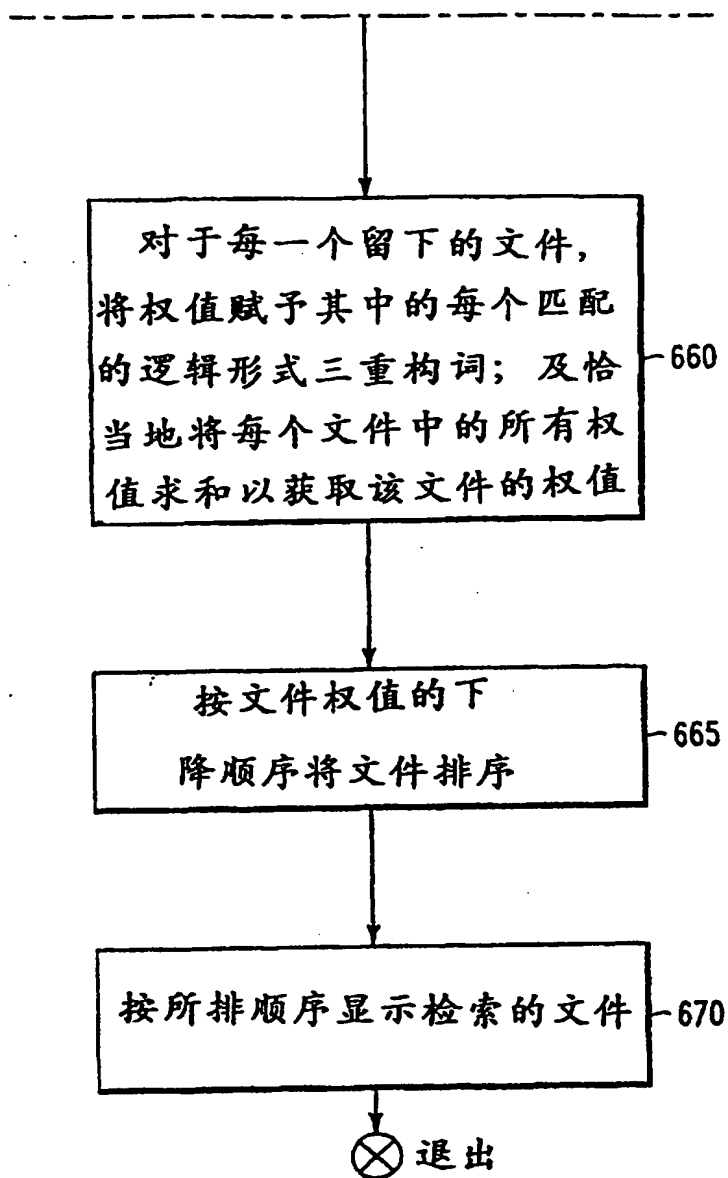


图 6

图 6B

NLP子程序

700

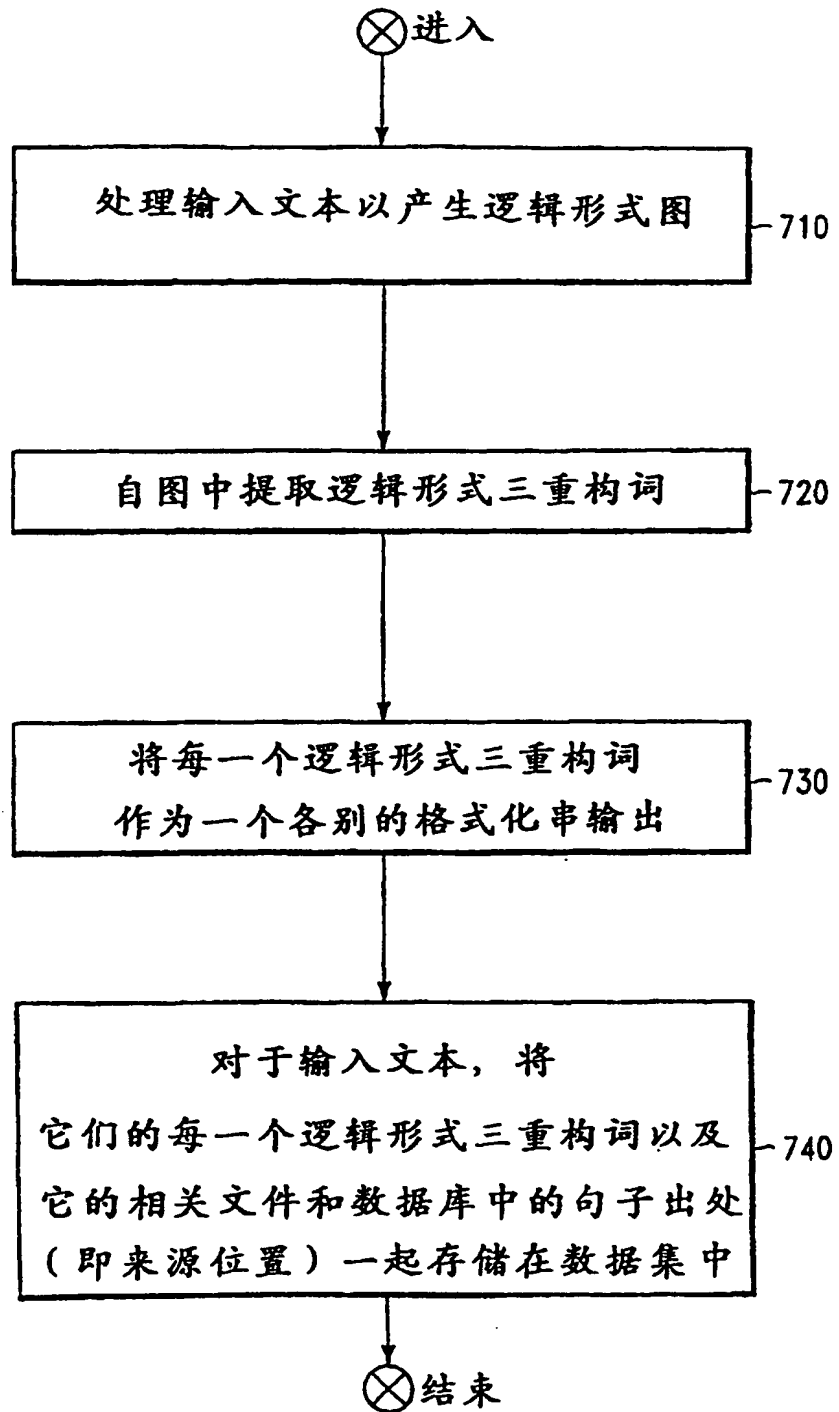


图 7

810 查询: HOW MANY HEARTS DOES AN OCTOPUS HAVE?

统计地检
索引的文件
820

DOCUMENT 1:包含ARTICHOKEHEARTS和OCTOPUS的配方
DOCUMENT 2:有关OCTOPI的冠词
DOCUMENT 3:有关DEER的冠词

NLP

逻辑形式三重构词

查询

HAVE-Dsub-OCTOPUS
HAVE-Dobj-HEART
HEART-Ops-MANY
830

DOCUMENT 1

HEART-Nadj-ARTICHOKE
FRY-Dobj-OCTOPUS
840

与查询的三重
构词无匹配

消除
DOCUMENT 1

DOCUMENT 2

✓HAVE-Dsub-OCTOPUS
✓HAVE-Dobj-HEART
HAVE-Dsub-OCTOPUS
850

与查询的三重
构词有2个匹配

全部文件权值
=100+75=175

DOCUMENT 3

✓HAVE-Dsub-DEER
✓HAVE-Dobj-HEART
860

与查询的三重
构词有1个匹配

全部文件权值
=100

匹配的重构 词的关系类型	权值
Dobj	100
Dsub	75
Ops	10
Nodj	10

匹配的逻辑形
式三重构词加权表

图 8A

按全部
权值排序

文件呈现顺序:
DOCUMENT 2;
DOCUMENT 3

图 8B

800

图9A

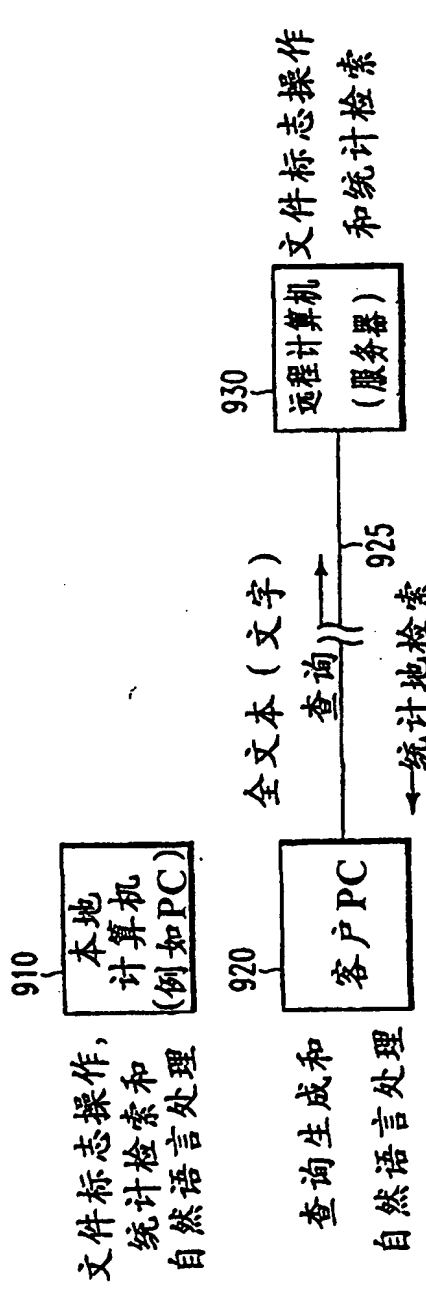


图9B

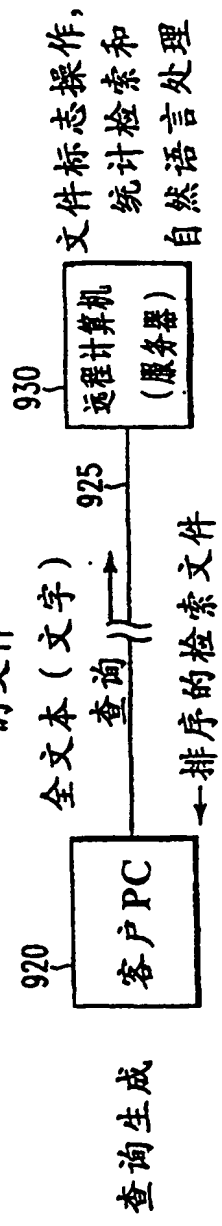


图9C

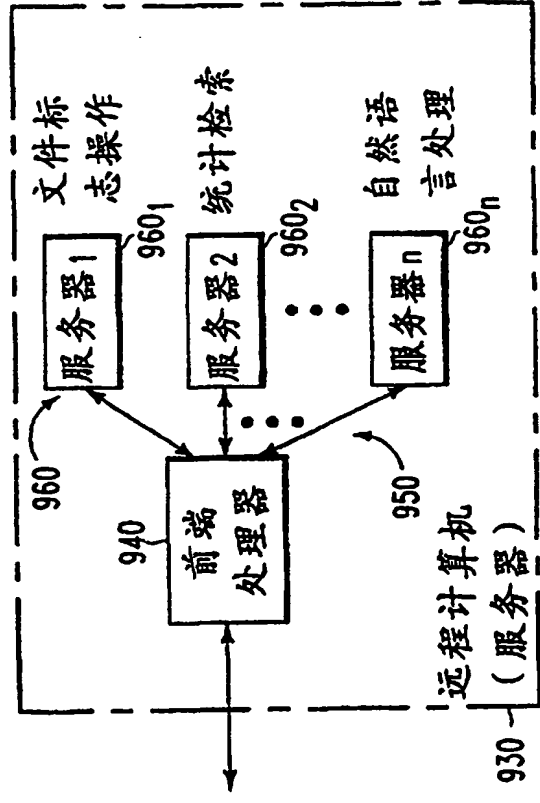


图9D

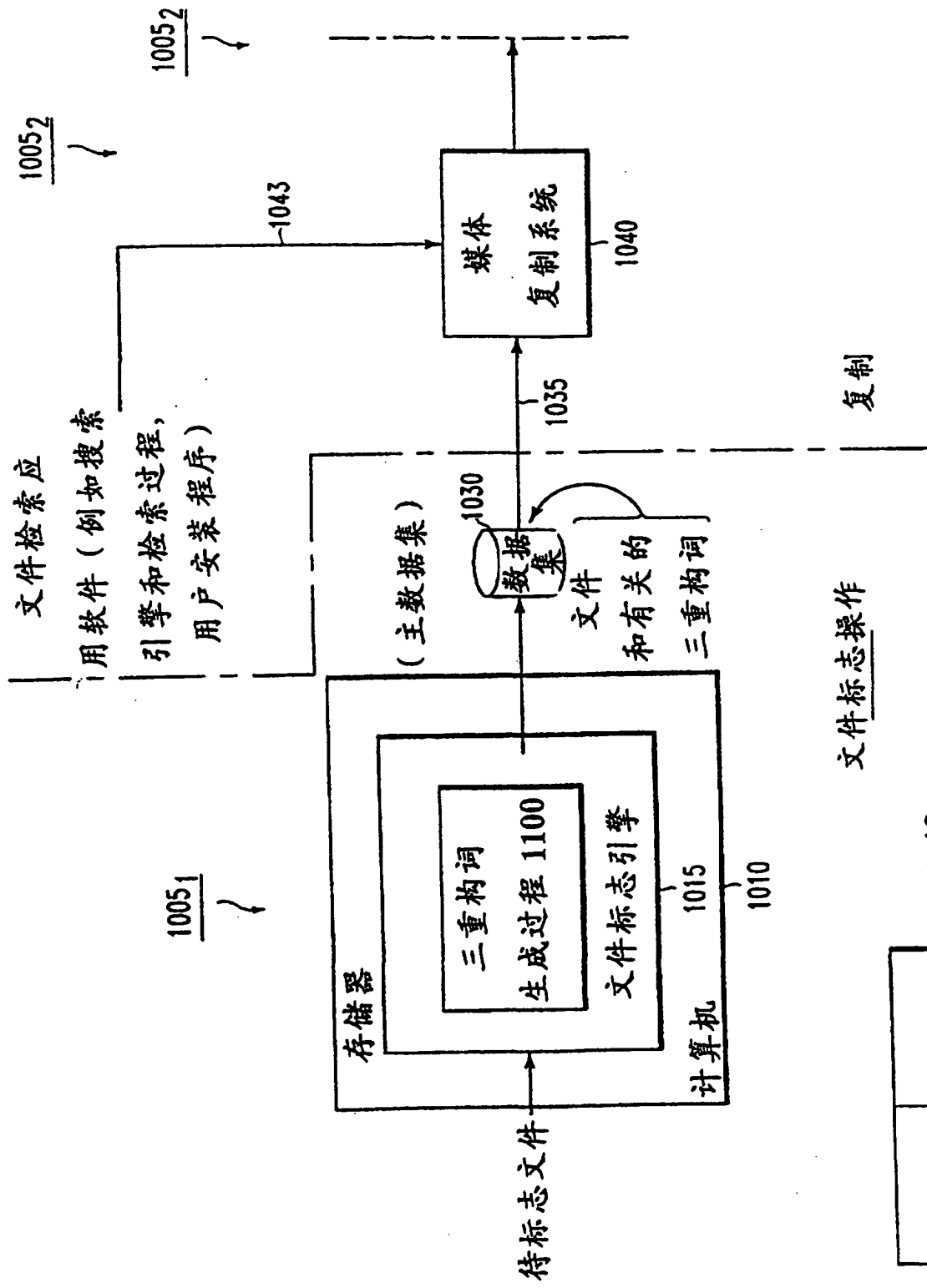


图 10A

图 10B

图 10

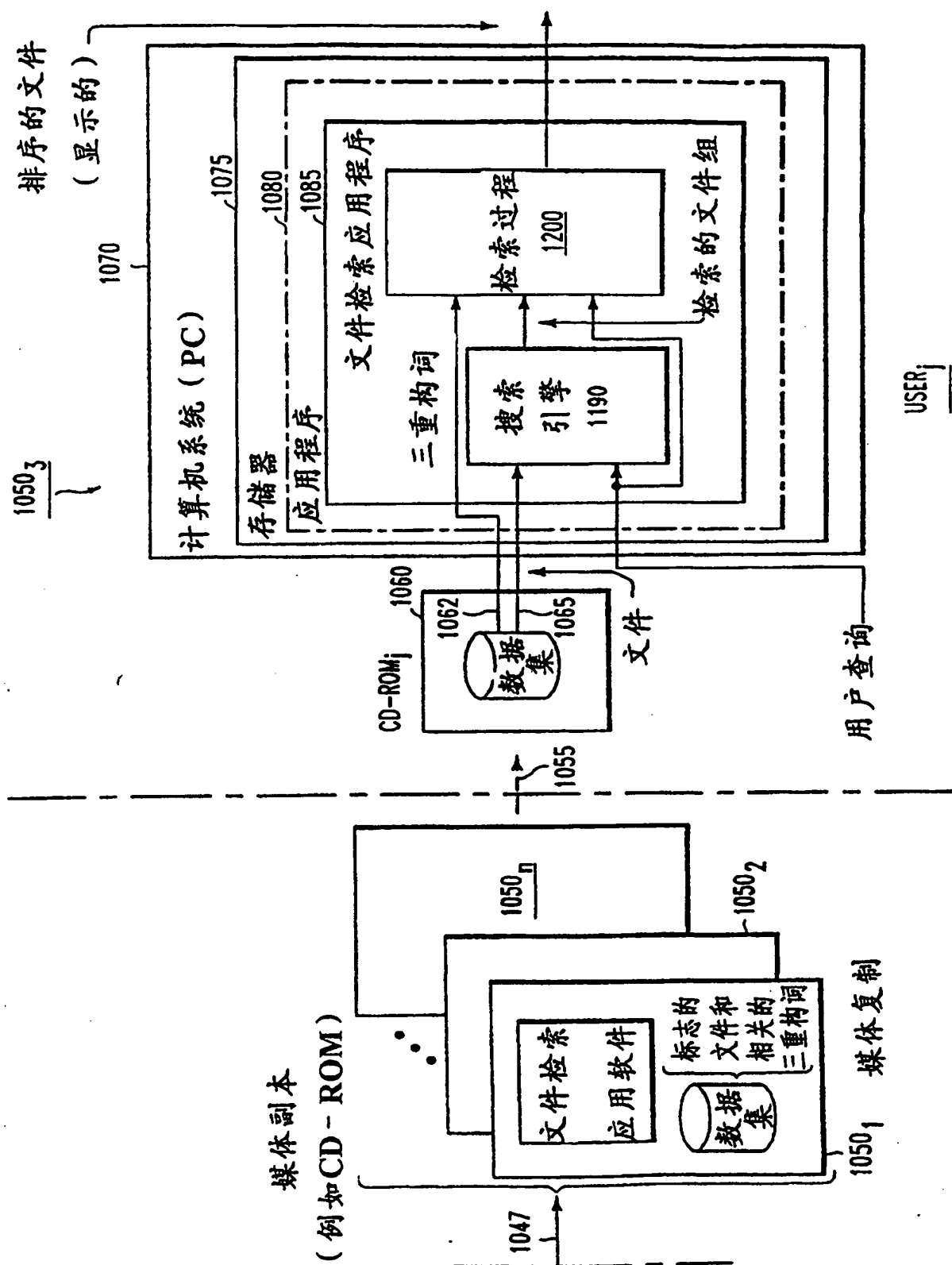


图 10B

图 11

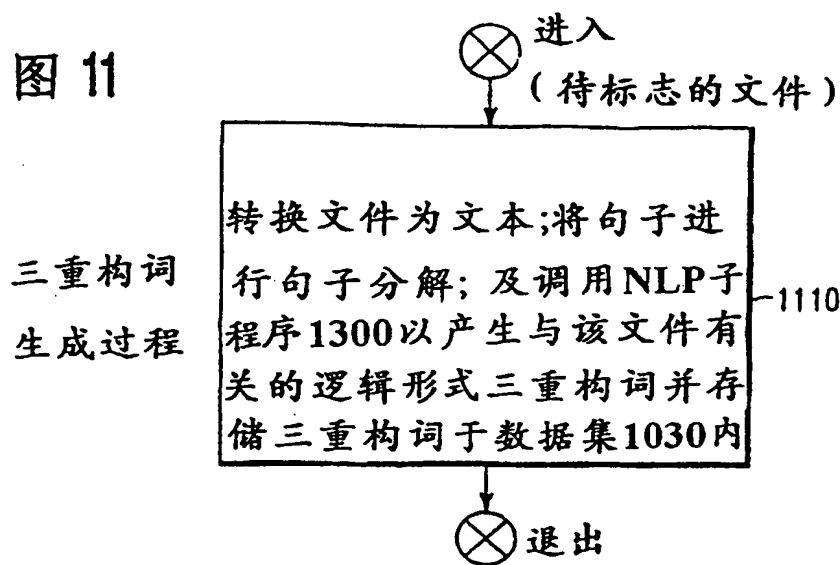


图 13A

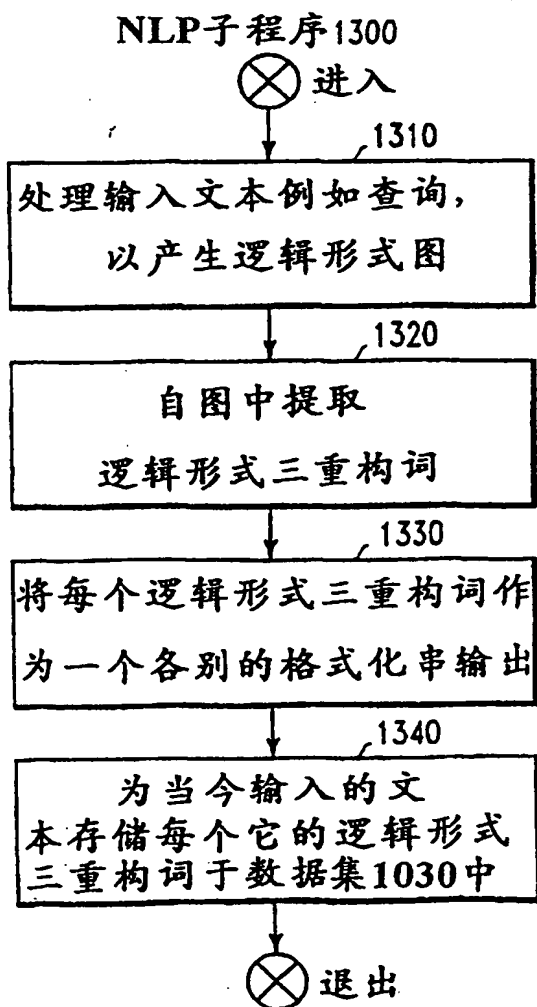


图 13B

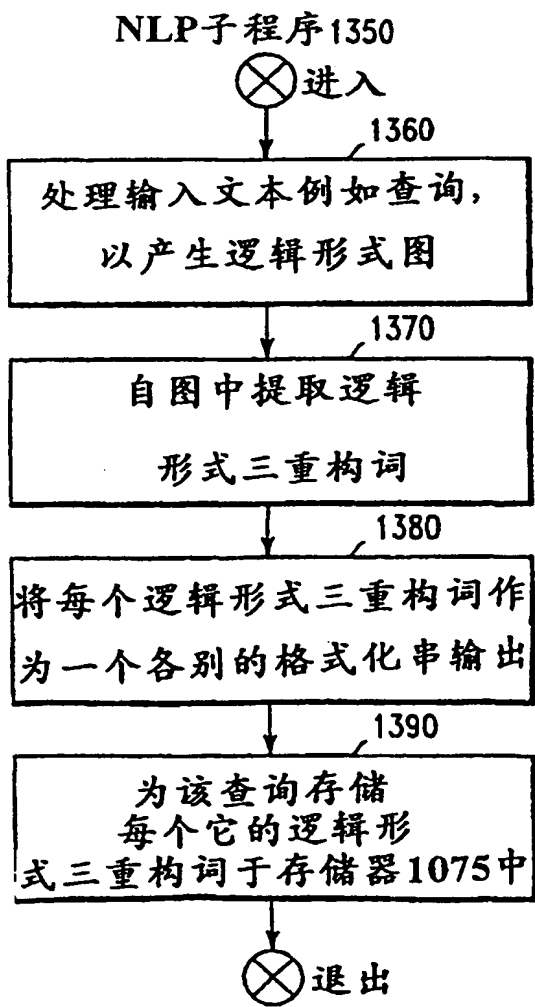
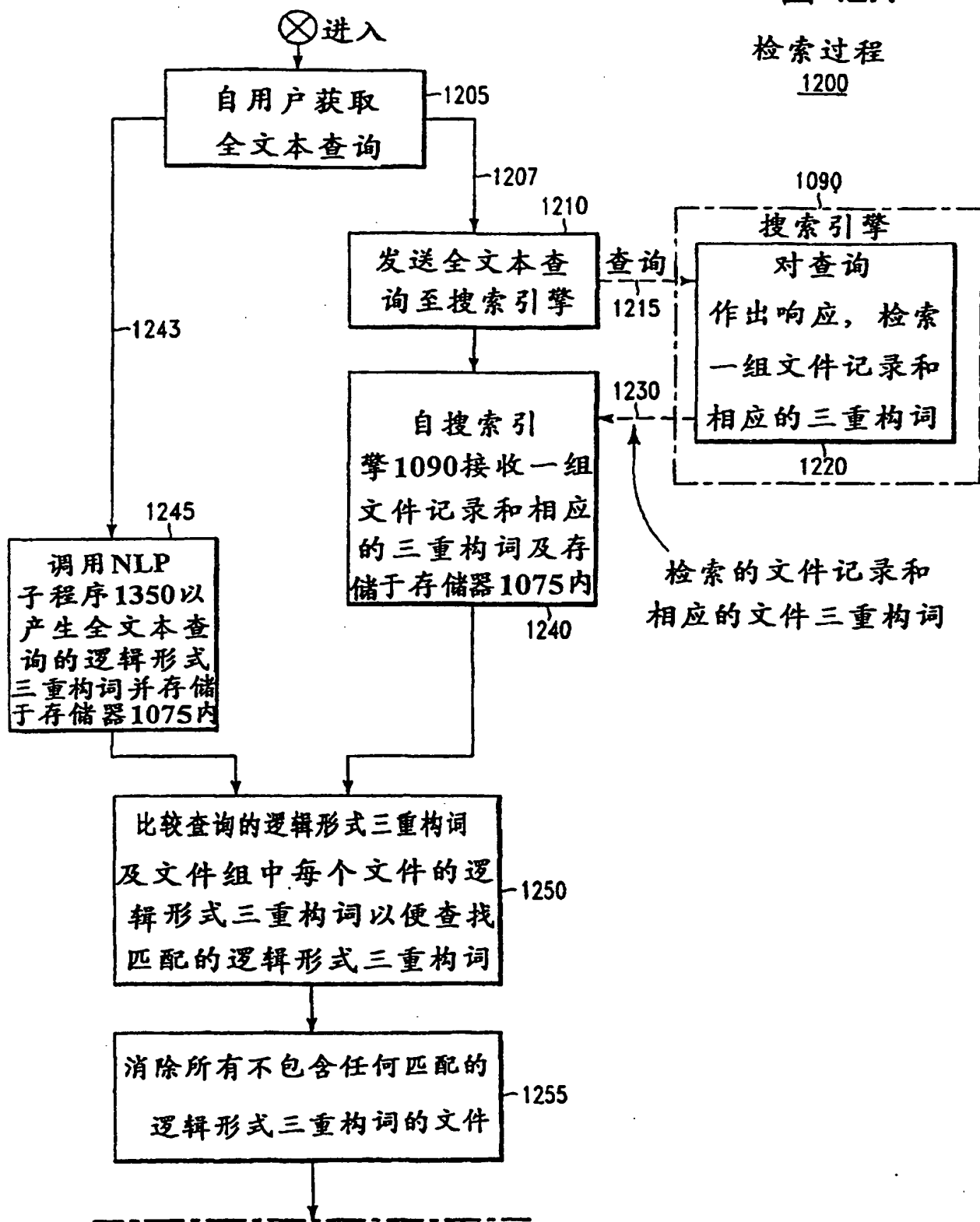


图 12A

检索过程
1200



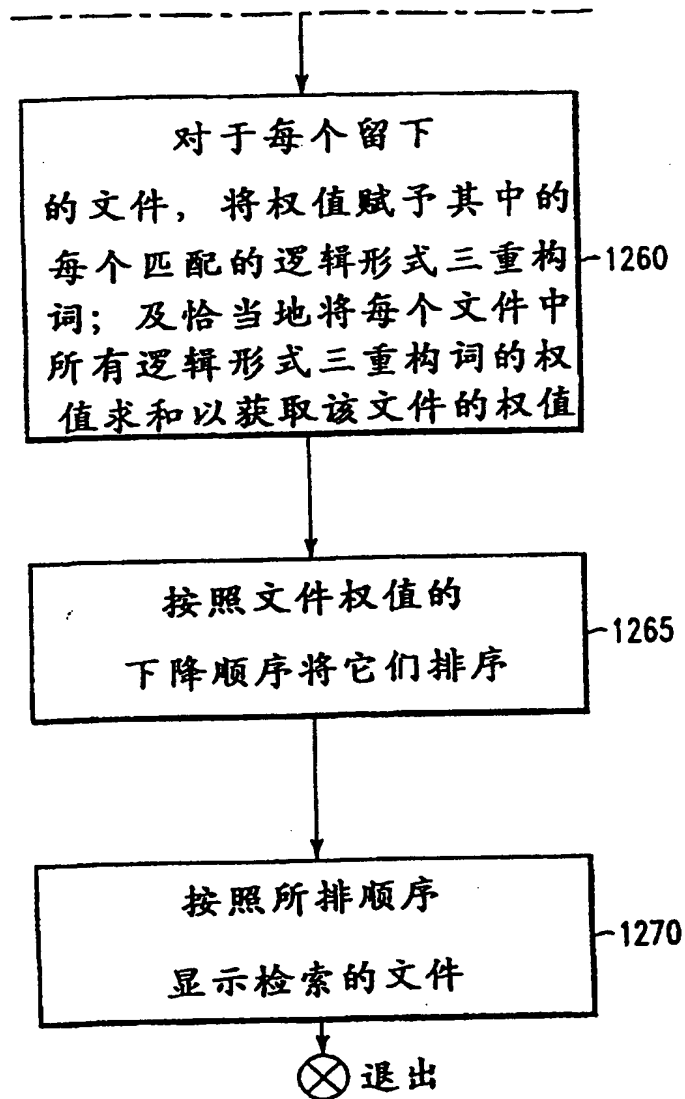
图
12A图
12B

图 12

图 12B